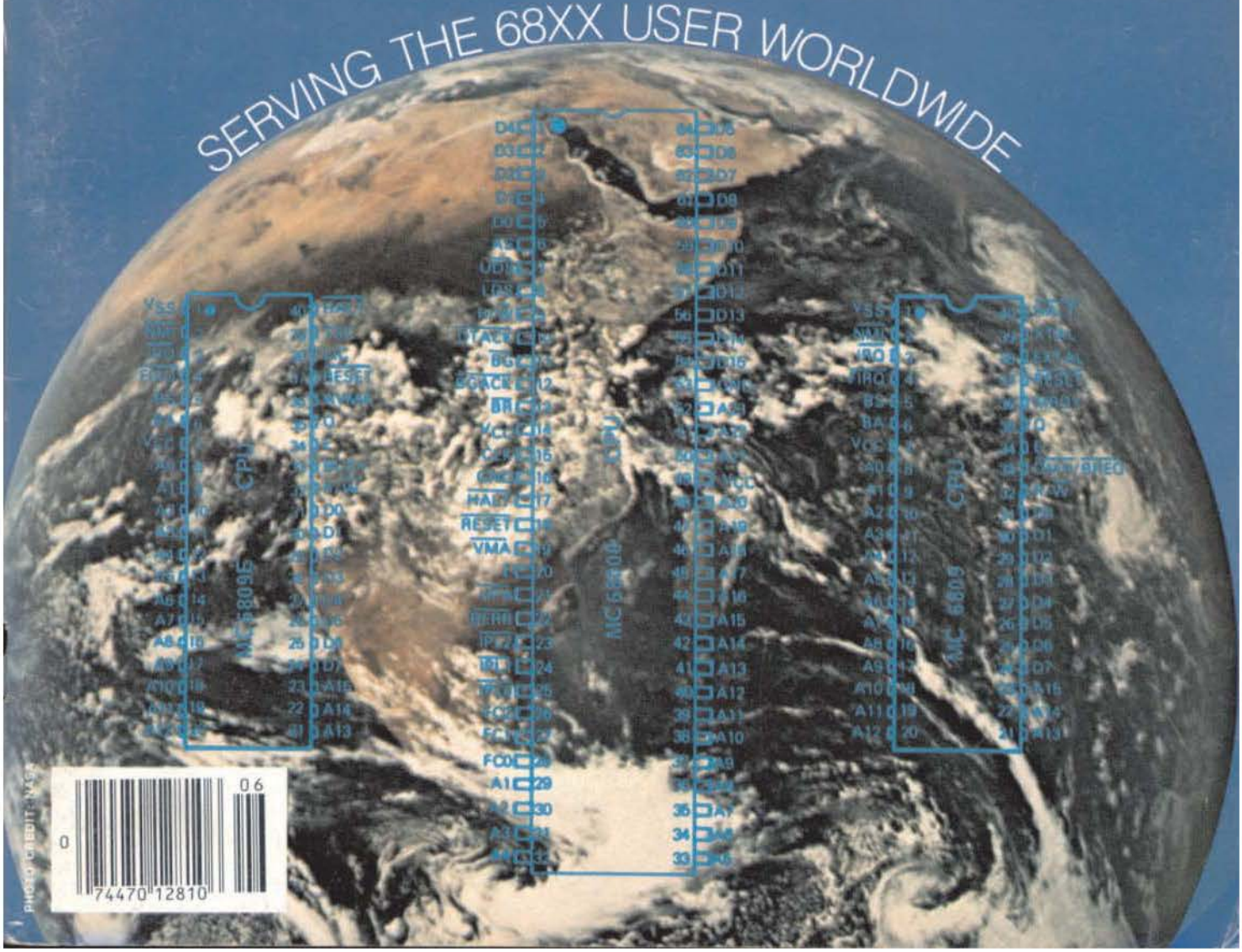
'68' MICRO JOURNAL

$2.95 USA

Australia	A $4.75	New Zealand	NZ $ 6.50
Singapore	S $9.45	Hong Kong	H $23.50
Malaysia	M $9.45	Sweden	30:-SEK

VOLUME VI ISSUE VI • Devoted to the 68XX User • June 1984

"Small Computers Doing Big Things"



0 74470 12810 06

WE DON'T PLAY GAMES

X-12+ A SERIOUS COMPUTER IN A DESKTOP PACKAGE

Multiprocessor Technology - Combination of 8, 16 and 32 bit types

1.0 Megabyte Memory - Insures no limitation on programs

"Winchester" Disk System - Fast response, large storage capacity

UniFlex* Operating System - The standard of comparison

Hardware Floating Point - Unmatched speed in a small system

Up to Three Terminals - Instant expansion

*Trademark of Technical Systems Consultants

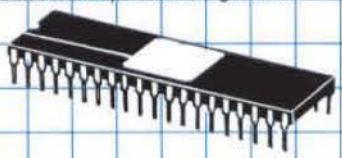
SWTPC

SOUTHWEST TECHNICAL PRODUCTS CORPORATION
219 W. RHAPSODY
SAN ANTONIO, TEXAS 78216 (512) 344-0241

MICROWARE'S OS-9 IS NUMBER ONE.

OS-9 NOW HAS THE LARGEST USER COMMUNITY

More users now run OS-9 on their 6809 computers than all other operating systems combined. This outstanding success story was no accident — it's due to OS-9's technical excellence backed up by outstanding Microware support. OS-9's Unix-type architecture and totally modular design gives your computer more power and versatility. OS-9 also gives you more possibilities for customization so you can tailor your system exactly to your needs. And aren't flexibility and performance the reasons you chose a 6809 computer to begin with?



OS-9 HAS BEEN CHOSEN BY OVER 50 6809 SYSTEM MANUFACTURERS

OS-9 is now offered as a standard operating system by almost every 6809 system manufacturer, and has been designed into an amazing variety of dedicated systems and products including personal and business computers, process control systems, data and telecommunications systems, and more. In all, over 50 companies and organizations have obtained OS-9 distribution licenses including such well-known names such as General Motors, NASA, Fujitsu, Western Electric, Motorola, Sykes Datatronics, Eastman Kodak, Thomson-CSF, and Tandy Corp.

OS-9 GIVES YOU A SOFTWARE BASE TO BUILD ON

Whatever your application, OS-9 speaks your language! Microware offers BASIC09, an Extended/Structured Basic, a complete C Compiler, a full ISO Pascal Compiler, the ANSI Standard CIS Cobol Compiler, plus Relocatable Macro Assembler. These high performance programming languages are all fully implemented and deliver unmatched performance and outstanding features. Additionally, OS-9 compatible applications packages such as word processors, screen editors, spreadsheets, business software, and utilities are offered by a rapidly growing number of third-party software houses.

PLUS OUTSTANDING MICROWARE SUPPORT: WE KEEP IN TOUCH WITH YOU

Even when you have the best software and documentation, there can be times when you need questions answered. That's why Microware is committed to giving OS-9 users the best possible **personalized** service. Here are some of the ways we deliver solid support:

- A Software Support Hotline for direct access to our technical staff
- "Pipelines", our free quarterly newsletter
- OS-9 User Seminars, the annual OS-9 community gathering
- a liberal update policy for new releases

Microware does business on a person-to-person basis. When you call you'll find yourself speaking with someone who's both knowledgeable and genuinely interested in helping.

YOU CAN COUNT ON OS-9 NOW AND IN THE FUTURE

Microware is not standing still — we're firmly committed to continuing support for the 6809 and we will continue to introduce exciting new software products for the 6809. We will soon announce OS-9/68000 and programming languages for the 68000 which will be upward compatible with 6809 versions, so if and when you are ready for the 68000 your OS-9 software can go with you.

MICROWARE®

Microware Systems Corporation
Box 4865 • Des Moines, IA 50304
515-279-8844 Telex 910-520-2535

OS-9 and Basic09 are trademarks of Microware and Motorola. Unix is a trademark of Bell Labs.

OS-9: BETTER BY DESIGN

'68' MICRO JOURNAL

Portions of the text for 68 MICRO JOURNAL was prepared using the following furnished hard/software.

COMPUTERS-HARDWARE
Southwest Technical Products
219 W. Rhapsody
San Antonio, Texas 78216
S09-5/8 DMF disk-CDS1-8212W-Sprint 3 Printer

GIMIX Inc.
1337 West 37th Place
Chicago, IL 60609
Super Mainframe-OS9-FLEX-Assorted Hardware

EDITORS-WORD PROCESSORS
Technical Systems Consultants, Inc.
111 Providence Road
Chapel Hill, NC 27514
FLEX-Editor-Processor

Great Plains Computer Company, Inc.
PO Box 916
Idaho Falls, ID 83401
STYLO-Mail Merge

Editorial Staff

Don Williams Sr.	Publisher
Larry E. Williams	Executive Editor
Tom E. Williams	Production Editor
Robert (Bob) Nay	Color Editor

Administrative Staff

Mary Robertson	Office Manager
Joyce Williams	Accounting
Carolyn Williams	Subscriptions
Penny Williams	File Management

Contributing Editors

Ron Anderson
Norm Commo
Peter Dibble
Dr. Theo Elbert
William E. Fisher
Dr. E.M. Pass

Special Technical Projects

Clay Abrams K6AEP
Tom Hunt

CONTENTS

Vol.VI,Issue VI June'84

FLEX USER Notes.....	8	Anderson
OS9 USER Notes.....	12	Dibble
Rumors & Such.....	16	DMW
C USER Notes.....	17	Pass
Reading Non-FLEX Disk Cont..	19	Frase
Disk Speed Measurement.....	28	Johnshoy
PC.....	30	Korte
Hacking The MAC.....	44	Wolf
Bit Bucket.....	35	
Help.....	48	
Classifieds.....	49	

Send All Correspondence To:

Computer Publishing Center
68 MICRO JOURNAL
5900 Cassandra Smith
PO Box 849
Hixson, TN 37343
615 842-4600

Copyrighted 1984 by
Computer Publishing Inc. (CPI)

68' Micro Journal is published 12 times a year by Computer Publishing Inc. Second Class Postage Paid ISSN 0194-5025 at Hixson, Tenn. and additional entries. Postmaster: send Form 3579 to 68' Micro Journal, PO Box 849, Hixson, Tennessee.

SUBSCRIPTION RATES
USA
1-Year $24.50 2-Years $42.50 3-Years $64.50
FOREIGN
See Page 60

Items Submitted for Publication

Articles submitted for publication should be accompanied by the authors **full name, address, date and telephone number.** It is preferred that articles be submitted on either 5 or 8 inch diskette in TSC Editor format or STYLO format. All diskettes will be returned.

The following TSC Text Processor commands **ONLY** should be used (due to our proportional processor): .sp space, .pp paragraph, .fi fill and .nf no fill. Also please do not format within the text with multiple spaces. The rest we will enter at time of editing.

STYLO commands are all acceptable except the .pg page command, we print edited text files in continous text.

All articles submitted on diskettes should be in TSC FLEX™ format, either FLEX2 6800, or FLEX9 6809 any version.

If articles are submitted on paper they should be on white 8X11 bond or better grade paper. No hand written articles (hand written or drawn art accepted). All paper submitted articles will be photo reproduced. This requires that they be typed or produced with a dark ribbon (no blue), single spaced and type font no smaller than 'elite' or 12 pitch. Typed text should be approximately 7 inches wide (will be reduced to column width of 3 1/2 inches). **Please use a dark ribbon!**

All letters to the editor should also comply with the above and bear a signature. Letters of 'gripes' as well as 'praise' are solicited. We attempt to publish all letters to the editor verbatim, however, we reserve the right to reject any submission for lack of 'good taste'. We reserve the right to define what constitutes 'good taste'.

Advertising: Commercial advertisers please contact the 68 Micro Journal advertising department for current rate sheet and requirements.

Classified: All classified must be non-commercial. Maximum 20 words per classified ad. Those consisting of more than 20 words should be figured at .35 cents per word. 20 words or less $7.50 minimum, one time, paid in advance. No classified ads accepted by telephone.

GIMIX HAS THE 6809 SYSTEM TO SUIT YOUR NEEDS

HARDWARE

All systems feature the **GIMIX CLASSY CHASSIS**; with a ferro-resonant constant voltage power supply, gold plated bus connectors, and plenty of capacity for future expansion.

Static **RAM** and double-density **DMA** floppy disk controllers are used exclusively in all systems.

All systems are guaranteed for 2 MHz operation and include complete hardware and software documentation, necessary cables, filler plates, etc.

Systems are assembled using burned-in and tested boards, and all disk drives are tested and aligned by **GIMIX**.

You can add additional components to any system when ordering, or expand it in the future by adding **RAM, I/O**, etc.

GIMIX lets you choose from a wide variety of options to customize your system to your needs.

SOFTWARE

All **OS-9/FLEX** systems allow you to software select either operating system.

Also included is the **GMXBUG** monitor and, in systems with 128K or more of RAM, **GMX-VDISK** for **FLEX**.

All **GIMIX OS-9** systems include **Microware's Editor, Assembler, Debugger, Basic09,** and **Runb**; and the GMX versions of **RMS** and **DO** for **OS-9**.

All **GIMIX** versions of **OS-9** can read and write RS color computer format OS-9 disks, as well as the Microware/GIMIX standard format.

New and exclusive with **OS-9 GMX III** systems is the **GMX OS-9 Support ROM**, a monitor for **OS-9** that includes memory diagnostics and allows the system to boot directly from either hard disk or floppy.

A wide variety of languages and other software is available for use with either **OS-9** or **FLEX**.

OS-9 GMX III/FLEX SYSTEMS (#79)

The **#79** super system now includes (in addition to the above): the **GMX 6809 CPU III**, a **256K CMOS Static RAM Board (#72)**, and a **3-port Intelligent Serial I/O Processor (#11)**.

The **GMX 6809 CPU III** can perform high-speed DMA transfers from memory to memory and uses memory attributes and illegal instruction trapping to protect the system and users from program crashes. If a user program crashes, only that user is affected; other users are unaware of the problem.

The 3-Port Intelligent Serial I/O Board (#11) significantly reduces system overhead by handling routine I/O functions; freeing the host CPU for running user programs. This improves overall system performance and allows user terminals to be run at up to 19.2K baud.

with dual 40 track DSDD drives	**$5998.79**
with dual 80 track DSDD drives	**$6198.79**
with **#88** dual 8" DSDD drive system	**$7698.79**
with **#90** 19MB Winchester subsystem and one 80 track	**$8898.79**
with a 47MB Winchester subsystem and one 80 track	**$10,898.79**
with a 47MB plus a 6MB removable pack Winchester subsystem and one 80 track drive	**$12,398.79**

TO ORDER BY MAIL: SEND CHECK OR MONEY ORDER OR USE YOUR VISA OR MASTER CHARGE. Please allow 3 weeks for personal checks to clear. U.S. orders add $5 handling if order is under $200.00. Foreign orders add $10 handling if order is under $200.00. Foreign orders over $200.00 will be shipped via Emery Air Freight COLLECT, and we will charge no handling. All orders must be prepaid in U.S. funds. Please note that foreign checks have been taking about 8 weeks for collection so we would advise wiring money, or checks drawn on a bank account in the U.S. Our bank is the Continental Illinois National Bank of Chicago, 231 S. LaSalle Street, Chicago, IL 60693, account #73-32033.

BASIC-09 and OS-9 are trademarks of Microware Systems Corp. and MOTOROLA, Inc.
FLEX and UniFLEX are trademarks of Technical Systems Consultants, Inc.
GIMIX, GHOST, GMX, CLASSY CHASSIS, are trademarks of GIMIX, Inc.

OS-9 GMX I / FLEX SYSTEMS #49

The **#49** systems include 64KB static RAM, #05 CPU, #43 2 port serial board.

with dual 40 track DSDD drives	**$3998.49**
with dual 80 track DSDD drives	**$4198.49**
with **#88** dual 8" DSDD drive system	**$5698.49**
with **#90** 19MB Winchester subsystem and one 80 track	**$5898.49**

OS-9 GMX II / FLEX SYSTEMS #39

The **#39** systems include 128KB static RAM, #05 CPU, #43 2 port serial board.

with dual 40 track DSDD drives	**$4498.39**
with dual 80 track DSDD drives	**$4698.39**
with **#88** dual 8" DSDD drive system	**$6198.39**
with **#90** 19MB Winchester subsystem and one 80 track	**$7398.39**

GIMIX DOES NOT GUARANTEE PERFORMANCE OF ANY GIMIX SYSTEMS, BOARDS OR SOFTWARE WHEN USED WITH OTHER MANUFACTURERS PRODUCT.

EXPORT MODELS: ADD $30 FOR 50Hz. POWER SUPPLIES.

GIMIX, Inc. reserves the right to change pricing, terms, and products specifications at any time without further notice.

ALL PRICES ARE F.O.B. CHICAGO

Contact GIMIX for price and availability of UniFLEX and UniFLEX GMXIII Systems.

NOTE on all drive systems: Dual 40 track drives have about 700KB of formatted capacity; dual 80's about 1,400KB; dual8" about 2,000KB. The formatted capacity of hard disks is about 80% of the total capacity.

Want to expand your system to a megabyte of Static RAM and 15 users?

Simply add additional memory and I/O boards. Your **GIMIX** system can grow with your needs. Contact us for a complete list of available boards and options.

#72 256KB CMOS STATIC RAM board with battery back up	**$1898.72**
#64 64KB CMOS STATIC RAM board with battery back up	**$528.64**
#67 64KB STATIC RAM board	**$478.67**
#11 3 port intelligent serial I/O board	**$498.11**
#43 2 port serial I/O board	**$128.43**
#42 2 port parallel I/O board	**$88.42**
#95 cable sets (1 needed per port), specify board	**$24.95**

TRADE UP YOUR CoCo!

GIMIX will allow you up to $1100.00 credit toward the purchase of any GIMIX system when you trade-in your working Color Computer, peripherals, and original software. The trade-in value is limited to 110% of the RADIO SHACK™ list price at the time your order is placed. You pay the freight. This offer is good only in the Continental U.S.; is limited to the first 100 orders; and expires on 9/30/84. Only one trade-in per customer.

GIMIX INC.
1337 WEST 37th PLACE
CHICAGO, ILLINOIS 60609
(312) 927-5510 • TWX 910-221-4055

©1984 GIMIX, INC. 4-84

Microware presents 4 new OS-9 software packages.

1 LEVEL II PRINT SPOOLING SYSTEM

This versatile package gives your OS-9 Level Two System a complete print spooling management capability for time-sharing applications. Features of the spooling system are:

- *Handles up to seven independent spooling devices and queues with "print on first available device" feature.*
- *Prints large block header pages between listings with date, time, user name and job name.*
- *Multiple listing copy option.*
- *Complete forms change capability for each job and device.*
- *Prints formatted or unformatted listings.*
- *Status command displays print queues and status.*
- *User can kill or change priority of queued jobs.*

*Available **only** for OS-9 Level Two Systems.*

Suggested List Price: $150.00 Manual Only: $15.00

2 RMA RELOCATABLE MACRO ASSEMBLER

At last — a full feature relocatable macro assembler and linkage editor for OS-9. RMA permits sections of assembly language programs to be independently assembled to "relocatable object files". The linkage editor takes any number of program sections and/or library sections and combines them into a single executable OS-9 memory module. Global data (including indexed and direct addressing modes) and program references are automatically resolved in the process. The macro facility permits commonly used statement sequences to be defined, then used within the program with appropriate parameter substitution. RMA also supports conditional assembly and library source files.

Suggested List Price: $200.00 Manual Only: $20.00

3 OS-9 FILE HANDLER TOOLBOX

Introducing a special toolbox for OS-9 users who do a lot of file manipulation! A collection of 12 useful OS-9 command programs; Most can be used as "filters" using OS-9 pipeline facilities. Included are:

***D** — unformatted directory listing with "wild card" matching*
***Compress** — does character compression on text files.*
***Expand** — restores a "compressed" file to the original state.*
***Split** — breaks a file into smaller files.*
***Space** — indents lines with optional spacing between lines.*
***Code** — decodes any key on a keyboard to hex.*
***Qsort** — quick sort for small files, directories, etc.*
***Pr** — versatile formatted file printing utility.*
***Tr** — transliterates text pattern to substitution pattern.*
***Grep** — searches file for a pattern and prints matching lines.*
***Xmode** — same "tmode" except changes are made to the device descriptor.*
***Count** — counts words, lines, or characters within a text file.*

Suggested List Price $85.00

4 ENTERTAINMENT PACK I

A collection of games and other interesting programs that are not only entertaining but serve as good instructional examples of Basic09 programming techniques. All programs include complete Basic09 source files and can be easily edited to run on standard alphanumeric or graphics terminals.

***Blkjak** — A Vegas-rules blackjack game.*
***Clk** — graphical display of a wall clock on your terminal.*
***Dogs** — Greyhound racing with simulated graphics.*
***Eliza** — Basic09 version of the famous artificial intelligence simulation of natural language dialogue with a psychiatrist.*
***Haiku** — Program that creates original "haiku" prose.*
***Quest** — a mini-"Adventure" game.*
***Rats** — find your way out of a computer-generated maze — from a rat's point of view.*
***Towers** — a graphical display of the solution to the "Tower of Hanoi" puzzle.*

Suggested List Price: $85.00

MICROWARE.

Microware Systems Corporation
P.O. Box 4865 • Des Moines, IA 50304
515-279-8844 • Telex 910-520-2535

OS-9 and Basic09 are trademarks of Microware and Motorola
Unix is a trademark of Bell Labs

FLEX™ USER NOTES
THE 6800-6809 BOOK

By: Ronald W. Anderson
As published in 68 MICRO JOURNAL™

The publishers of 68 MICRO JOURNAL are proud to announce the publication of Ron Anderson's **FLEX USER NOTES,** in book form. This popular monthly column has been a regular feature in 68 MICRO JOURNAL SINCE 1979. It has earned the respect of thousands of 68 MICRO JOURNAL readers over the years. In fact, Ron's column has been described as the 'Bible' for 68XX users, by some of the world's leading microprocessor professionals. Now all his columns are being published, in whole, as the most needed and popular 68XX book available. Over the years Ron's column has been one of the most popular in 68 MICRO JOURNAL. And of course 68 MICRO JOURNAL is the most popular 68XX magazine published.

As a SPECIAL BONUS all the source listing in the book will be available on disk for the low price of: FLEX™ format only — 5" $12.95 — 8" $16.95 plus $2.50 shipping and handling, if ordered with the book. If ordered separately the price of the disks will be: 5" $17.95 — 8" $19.95 plus $2.50 shipping and handling.

Listed below are a few of the **TEXT** files included in the book and on diskette.

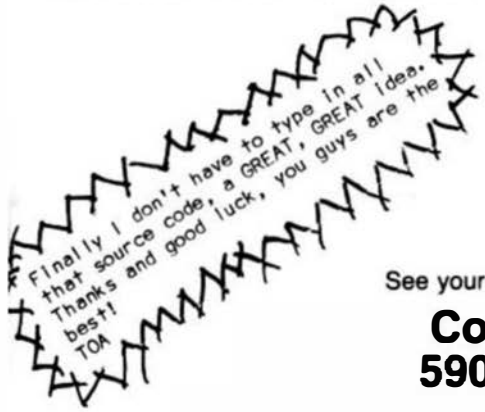
All TEXT files in the book are on the disks.

LOGO.C1	File load program to offset memory — ASM PIC
MEMOVE.C1	Memory move program — ASM PIC
DUMP.C1	Printer dump program — uses LOGO — ASM PIC
SUBTEST.C1	Simulation of 6800 code to 6809, show differences — ASM
TERMEM.C2	Modem input to disk (or other port input to disk) — ASM
M.C2	Output a file to modem (or another port) — ASM
PRINT.C3	Parallel (enhanced) printer driver — ASM
MODEM.C2	TTL output to CRT and modem (or other port) — ASM
SCIPKG.C1	Scientific math routines — PASCAL
U.C4	Mini-monitor, disk resident, many useful functions — ASM
PRINT.C4	Parallel printer driver, without PFLAG — ASM
SET.C5	Set printer modes — ASM
SETBAS1.C5	Set printer modes — A-BASIC
	(And many more)

**Over 30 TEXT files included in ASM (assembler) — PASCAL — PIC (position independent code) TSC BASIC-C, etc.

NOTE: .C1,.C2, etc. = Chapter 1, Chapter 2, etc.

This will be a limited run and we cannot guarantee that supplies will last long. Order now for early delivery.



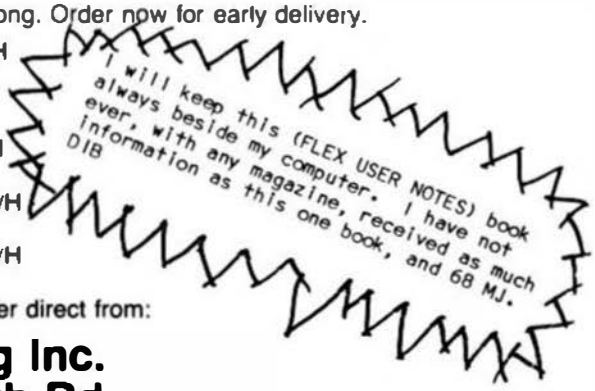
Foreign Orders Add $4.50 S/H

Softcover — Large Format

Book only: **$7.95** + $2.50 S/H

With disk: 5" **$20.90** + $2.50 S/H

With disk: 8" **$22.90** + $2.50 S/H



See your local S50 dealer/bookstore or order direct from:

Computer Publishing Inc.
5900 Cassandra Smith Rd.
Hixson, TN 37343
(615) 842-4601

™FLEX is a trademark of Technical Systems Consultants

$4,325
FOR A WORLD-CLASS SS-50 COMPUTER

Smoke Signal's VAR/68™ gives you:

- Fabled Chieftain performance that led the pack in tough Benchmark surveys
- Integrated, easy-to-use software that covers your *complete* business needs
- Proven reliability backed by our exclusive Endurance-Certification Program
- Extremely good looks and unsurpassed operator comfort

(2) Our Advance-Replacement program is yours for a low fixed charge. (3) You get instant diagnostic service by telephone. It's free. (4) Normal repairs are handled with super speed. (5) Software *and* hardware support are part of doing business with Smoke Signal.

TOTAL INTEGRATED SOFTWARE GIVES YOUR BUSINESS *SOLUTIONS* INSTEAD OF PROBLEMS

Powerful business application programs are ingeniously interlinked to give even untrained operators a quick, smooth upper hand. The VAR/68 is a joy for first-time users, and an unprecedented productivity tool for a*nyone* who wants new dimensions of control over critical business matters.

This screen tells part of the story:

$4,325: A PRICE CALCULATED TO GET YOU HOOKED ON THIS BLOCKBUSTER SS-50

That price buys you a VAR/68 computer with multi-user, multi-tasking capabilities, and an ergonomically designed terminal. You get 128K RAM—expandable to 1mb. Eight serial ports, up to 16 if desired. Two parallel ports—and more are available. Plus a long list of other impressive capabilities.

Smoke Signal's experience allows us to offer OS-9 and other UNIX-like, and multi-user operating systems.

The styling is completely new—fashioned for the utmost in operator comfort. And it's remarkably compact. VAR/68 is a combination of great performance and good looks demanded by the office of today.

VAR/68 IS TOUGH, BUT SMOKE SIGNAL GIVES YOU *EXTRA* PROTECTION

(1) Your new computer is Endurance-Certified before delivery. That's an exclusive quality-assurance process that guarantees perfect operations from day one.

GET A BIG DISCOUNT ON YOUR INITIAL ORDER

Most re-sellers can save up to 42 percent—even on small orders. Smoke Signal's price schedule is a powerful profit-maker for dealers of almost every description.

CALL SMOKE SIGNAL OR WRITE FOR MORE INFORMATION ON THE VAR/68 COMPUTER FAMILY

VAR/68™
BY SMOKE SIGNAL
The computer designed for your specialized application software

SMOKE SIGNAL
Products and Support for VARs

31336 Via Colinas • Westlake Village, CA 91362-3984 • (818) 889-9340

TMP (Total Management Planning) OS-9 SOFTWARE

THESE OFF-THE-SHELF APPLICATION PACKAGES HAVE CHANGED THE PICTURE FOR PROFESSIONAL SOFTWARE USERS

Here are the comprehensively supported OS-9 application software packages that discerning professionals are selecting today. Each one runs stand-alone or in combination with other TMP packages.

TMP/CALC is a *new-generation* "Electronic Spreadsheet" package that goes a quantum leap beyond ordinary Calc software. Superior speed, and extra features like Dynamic Calculations and Dynamic Overlay, mean you can change a cell *anywhere* on the spreadsheet and get automatic updates wherever they belong. No more "wrap-around" since output to the printer is formatted. "Help" screens prompt without erasing data, and the rows and columns on any worksheet are limited *only* by available memory.
ONLY $250

TMP/Manager is a structured database manager, and is the ultimate driving force behind any total *Management/Marketing Planning* system. Its built-in select/sort/merge module — and optimized machine code — makes it superfast, and provides unusually nimble data manipulation characteristics. Other outstanding features are detailed prompting, easy identification of data fields, and an excellent security system.
ONLY $500

TMP FreeForm fills the void between a structured DBMS and word-processing; it's an "electronic index card" package with an endless list of valuable applications such as parts cataloging, look-up inventories and listings of *any* kind. Type anything on up to 32,000 electronic "cards" of up to 9 pages each that reside in "file drawers" (databases). Up to 13 key words can call each page.
ONLY $150

TMP/Front-End integrates any or all of the packages described above, and integrates TMP with word-processors, BASIC and other high-level languages.
NO CHARGE with TMP/Manager

TMP T.M.

TMP PACKAGES ARE EXCEPTIONALLY FRIENDLY. EACH COMES WITH MULTI-MEDIA TRAINING AND OUTSTANDING DOCUMENTATION

AUDIO CASSETTE TRAINING TAPES are provided with each TMP software package. These crisp, professionally produced training aids take a new user through "hands-on" exercises that instantly build confidence, and minimize reference to manuals.

CLEAR DOCUMENTATION is specially prepared for each TMP package. Remarkably readable because they are conversational in style and logically organized, this series of manuals *further* assures quick user comfort and productivity with TMP software. Includes sections on how to best utilize the system; integrating user written software with TMP packages, and other bonus topics.

VIDEOTAPE PROGRAMS are available to TMP Dealers. They dramatically take the viewer into real business environments for a close look at how a TMP software package is utilized. Video programs can be personalized for specific organizations. Programs may be specified for most popular tape formats.

THE BOTTOM-LINE IS THIS: Software is only as effective as its training and documentation. TMP packages are *incomparable* in both areas.

RUN TMP IN A SMOKE SIGNAL CHIEFTAIN™ COMPUTER TO EXPERIENCE THE KIND OF SPEED AND POWER YOU CAN EASILY AFFORD *TODAY*

Endurance Certified Chieftains are consistently among the fastest computers in CPU and I/O speeds according to the widely respected BENCHMARK REPORTS. That, combined with almost legendary reliability, makes Chieftains the logical choice in computers using OS-9 operating systems.

Configurations range from floppy disk systems to multi-user, multi-tasking Winchester hard disk systems with tape back-up; performance approaching mainframes at prices you can live with.

TMP software packages for most OS-9 systems and Chieftain computers are offered exclusively from Smoke Signal. Inquiries from non-Smoke Signal dealers are invited. Call (213) 889-9340.

SMOKE SIGNAL
Chieftain Computers

31336 Via Colinas • Westlake Village, CA 91362
(213) 889-9340

Flex User Notes

Ronald W. Anderson
3540 Sturbridge Court
Ann Arbor, MI 48105

Reader Response

My "Printer Pal" Art Weller wrote me after seeing last month's column (sneak preview) and I will quote his interesting and objective comments directly.

"I just finished reading the latest in the 'Anderson - Farnsworth' Feud' (just joking -- I realize it's a friendly argument) and decided to put my two cents worth in. First of all, I conclude that neither one of you can win for several reasons; (1) the ground rules haven't been established so that it can be determined when a 'win' has taken place, and (2) it's becoming clear that neither is going to be able to convince the other.

"There are complers and there are compliers. You and I have talked about this before and agreed that people who design and develop compilers do so with some rather specific objectives in mind. Pascal as an aid to teaching structure, Cobol for business oriented applications, etc., etc. As a result, we agree, there is no one 'best' compiler that will be optimum for whatever application may come along. Further, when considering a specific compiler for some specific application, the compiler 'loses points' to the extent that the programmer must write his own function (paraphrasing an Anderson comment about Forth). As you know, I'm very much impressed with the ease and clarity of Pascal programs (when compared to, say, BASIC), but Pascal is a rather poor choice for an application that involves a lot of high speed string manipulation." (NOTE: Al Jost of Dynasoft and I would probably debate that point. RWA) "I think you see where this logic is leading and I will quit here so I can get to the point.

"The point is that I really have to sympathize with Dan because there's no way to pin Ron down. If Dan makes some really telling point in the argument, Ron can respond instantly by switching to a different compiler that happens to hve been optimized to handle that particular problem.

"On the other hand, I can't help wonder if Dan realizes that what he has really done is to devise a rather specialized 'compiler' that just happens to have been tailored to his needs -- no wonder he's so happy with it. No matter that the interface with the user is crude, or that the language is not standardized, consisting mostly of library calls or that an assembler is used to interface with it -- it is performing the same functions for him that a more standardized HLL is performing for me. The statement that reveals this is 'Let us not compare primative assembly language to HLL's when I am advocating a much more sophisticated approach.' I translate that to mean that Dan is not advocating assembler language, but prefers something more sophisticated, eg. a HLL!

"Switching sides quickly, I have to point out that when Ron is evaluating a compiler, one of the criteria is the ease (or lack thereof) of introducing assembler language operations into the flow of the program. And of course this ranges from BASIC's clumsy POKEing of decimal numbers (ugh!) to the use of actual assembler mnemonics in the program listing." (RWA NOTE: as in "C") That is, despite the sophistication of the best available HLL's, there are still times when good old tedious assembler is the way to go.

"This brings me to the comment by Dr. Franz (making sure not to miss offending everyone!). Gee, I wish he hadn't used the word 'death'. It's true that 'main frame' computer programming evolved from 'machine code' to the present HLL, and 'better' languages are still being devised. But it should be noted that machine code programming of main frame computers was at a time when computing 'power' was on the order of (or even less than) your Kim-1 and the evolution of more sophisticated programming techniques paralleled hardware complexity vs cost. That evolution is still going on and just happens to include micros, but I really doubt that we'll see the complete demise of assembler programming or machine code, for that matter. (e.g. 'MOCONTROL = $3;' -- isn't the $3 value a 'machine code instruction' for the 6850 serial interface? Probably the FDC1771 disk controller would provide much better examples).

"If you don't mind my restating the point I tried to make in an earlier letter, let me tell you about an auto oil change. I usually do it myself, but sometimes weather doesn't permit so I go to a service station and on one occasion, while waiting, I happened to glance at my car on the lift just in time to see the 'mechanic' about to remove the drain plug with a pipe wrench! Of course that would ruin the plug so that the proper wrench wouldn't fit -- I actually screamed at him. Just as any mechanic, worth his salt, should have the proper tools and use them appropriately; it seems to me that, likewise, a competent programmer will also insist on a complete toolbox and carefully select the right language suitable to the application he is working on."

Gee, I think I ought to let Art write this column for a while. Thanks, Art for the objective viewpoint.

Dan Farnsworth's Reply

"I would like to compare the operation of an assembler to the operation of an automobile with a 4 speed transmission. Previous to your last column you could only think of assembler operations in low gear. However, last month you discovered Macro's and got yourself shifted into 2nd gear and showed some pretty good examples. The following will demonstrate the assembler working in high gear with Library routines."

(NOTE: I'm going to cry FOUL here. I didn't just discover Macros last month... RWA)

"I have a Floating Point BCD Math Package and from this I have extracted the Multiply routine (FPMULBCD). To solve the weight of the steel cylinder problem from last month, I would code the following: (SEE LISTING BCDWGT)

"This assembles into 347 bytes of code, about 1/3 of your code from last month and 1/6 of the code generated by PL/9."

FOUL FOUL FOUL

I was very careful to state that in order to compare oranges to oranges, I was including the four functions, not just the multiply!!!! (RWA)

"The answer is correct to .001 ounce and the execution time is 14 Ms. If you want a nice formatted output then another 300 bytes would be required and an input routine would add 140 bytes. This Floating Point Package has a range of numbers from plus or minus 10^-64 to 10^63 with 8 digits of accuracy on Add and Subtract, but drops to 6 digits for Mult & div. More or less accuracy can be changed by setting the PC to ay even number.

"Now let us shift into overdrive. I have in my computer an Eprom which contains a 10 digit bcd fixed point mathpack. This resides at $E800 and it is always loaded and ready to go. The coding is as follows: (SEE LISTING FPBCD).

"This assembles to 40 bytes of coe and 25 bytes of storage or 65 bytes total. The accuracy is 1/2 ounce and it takes 8 ms to run. If you want a formatted output, change the JSR MOVTOT to JSR MOPPRTO (move total and print it). If an input is needed then a call to GETNUM will put the number in the NUMBER buffer. If you need more accuracy then everything could be scaled by a factor of 10 or 100. If you wanted to make 100 calculations it would take from 10 to 12 bytes of code to fetch and store the radius and height in straight line code (a loop would be much less) making 1500 bytes compared to your 7.4K in PL/9.

"Ron, I don't think anyone would argue that assembled programs will run faster and use less memory than their equivelents written in an HLL. Our discussion has been "Is the effort worth the trouble". It is quite obvious that if you have to write in an area you have never done before and probably will never do again an HLL is the way to go. If you are going to write several programs in the same area and you take thetrouble to develop a library of subroutines or even better a run-time package, you will find that assembler is a "piece of cake.

"Last month I promised to include the source code for GETNUM an input routine for my BCD mathpack. (SEE GETNUM LISTING). Next month I will include PRBCD, a routine that will output a very nicely formatted number with leading zeros suppressed, commas inserted at the right places, and the decimal points lined up.

My Reply

I guess we are not very far apart at this point, Dan. Your assembler routines certainly look like my assembler routines. I was careful to make all my comparisons last time on the basis of having four functions and I/O included in the byte count, and your results are not spectacularly different than mine. I ran the calculation in the PL/9 program, and it ran (I am assuming your 8ms and 14ms are for a 2 mhz system) in 1.75 ms. You see, the binary math packages are QUITE a bit faster than the BCD. For money calculations with zero rounding errors, I agree the BCD is better. For scientific calculations where the input data may be a couple decimal places less accurate than the math package, I really don't care if I am off by one or two in the last place due to rounding errors.

Perhaps that result will once and for all kill the idea that compliers have to generate code that is slow to execute! If you write and tell me that your times are for 1 MHz, then the PL9 version is still more than twice as fast as your admittedly limited accuracy fixed point calculation, and four times as fast as the floating point BCD with the same accuracy as the PL/9 version.

It seems as though this debate has reached the point where there is little left to say. Your last paragraph seems to sum up what you have been saying. I can sum up my argument by saying that for the type of programming I do most of the time, a penalty of 2 in the size of the binary output code and little or no speed penalty, is a tiny price to pay for the ease of writing my programs in a higher level language. With EPROMS at $4 or so for a 2732, one of the instruments I design and program will cost an extra $8 in hardware, and several hundred dollars less in my programming time. That's a pretty good trade off. It wouldn't be so good if we were producing 10000 identical units, which would then cost an extra $80,000 in parts to save several hundred in my time. That has been my point all along.

Permit me just one more illustration. How many SS-50 computers are there out there? They continue to be sold, and I suppose now there might be some 15,000 around. Do you think any one piece of software (say a screen editor, for example, for $200) will ever sell more than about 1000 copies? If it does, those sales will be spread over a couple of years, what we would call the "life" of the product. Over that time, the company that wrote the software has to support it, advertize it, make copies, print manuals, pay postage, and pay a portion of the company's expenses for overhead (heat light, phone, rent). If the distributor sells it for $200, the supplier most likely wholesales it for about $130 or so. The cost of writing the software in assembler could easily hit 30% of the total revenue. Writing it in a higher level language could easily reduce that by a very substantial amount, perhaps to 10% of the total revenue. That might easily make the difference between a profit and a loss on that item.

When we talk about 10,000 or 100,000 total sales, thre is little doubt that Assembler is the way to go. Somewhere in that ballpark of 1000 however, the choice has to be to go with the HLL for any company that wants to stay in business.

Though I think the discussion has run its course, I'd be happy to publish further pieces of Dan's math package if he is willing to share them with us.

New Complier

Frank Hoffman, the author of CRASMB, the cross assembler(s) and several other useful software products, is putting the finishing touches on his K-BASIC complier at the time of this writing. I've known Frank for some time as a result of his sending me CRASMB for a review a couple years ago. A couple of months ago he asked me if I would write the scientific functions for his package. We agreed on a reasonable fee (this is my second software for profit venture) and I have just completed and tested the functions, and integrated them into his runtime package. I think in view of the circumstances someone else ought to do a full review after the package is released, but I do want to mention it, and indicate that it is very nearly compatible with TSC Extended BASIC at the source level. I have compiled several BASIC programs with minimal changes, and had them run correctly. The advantage, of course, of having a compiler that is compatible with an interpreter, is that the program may be developed using the interpreter, and then compiled after it is working. Suppliers of software written in BASIC, such as business software, Database systems, etc. will be able to avoid the frequent "chaining" of programs since the code is smaller and more can be held in memory at one time. Also the code will be harder for the user to "optimize" thus reducing problems for the supplier.

This was a "first test version" and of course I found a few problems in it, but they were all relatively simple to fix. I was impressed with the modularity of the runtime package, and I was able to code some of the standard scientific functions with little trouble using the stack for "scratchpad" memory for the required calculations. I think this is going to be a winner. Watch for a review of it soon.

Program Correctness Again

I've just been through a couple of long processes in getting a bug free version of a compiler or two. This discussion is not an attempt to pick on anyone in particular, and in fact the reason for it is that problems such as I am going to discuss are nearly universal among software suppliers. The problem is simply that of too many bugs in officially "released" software, and too long a time in getting them all removed.

I once received a "C" compiler with float and long capability. I found that I couldn't input a negative number via scanf(), the sign was simply discarded. printf() would output a negative number but it wouldn't be formatted as specified. The logical comparisons for long integers didn't work in general and the ones that did, didn't work in logical combinations with and and or, since true returned as 1 in one case and -128 in another case. Anding these together could never result in a non zero value. I found that the basic arithmetic didn't work. Doing a long integer multiply, multiples of 65536 were lost. The real arithmetic had problems too. For certain specific numbers, the result returned was zero. The supplier had implemented successive subtraction as a method for divide for integer numbers. Dividing 32767 by 2, for example, required 16384 passes through a loop, and took half a second!

Not all of the software suppliers and writers have problems that are that obvious. I've been going around for seven months with a supplier who, all this time has been advertizing his product and selling it. I have yet to be able successfully to compile the first program I tried to compile 7 months ago. As each problem was uncovered and reported, that problem was fixed (but not related ones). First try, the initializer feature didn't work for floats. Second try, it worked for positive valued of floats but not negative. Third try it wouldn't initialize an array of type float of static storage class.... It seemed that each time the author would only go so far as to fix the specifically reported problem and go no further into checking even within the same feature to see if any more problems were present. Currently I know of only one remining problem, and it is not a very serious one. The supplier in question has been very good about supplying updates to customers and in being able to say that "all KNOWN bugs are fixed" at any point in time.

Understand that I am not talking here about suppliers who send out preliminary or pre-release versions of their software for testing, but releases that are advertized as being finished and free of known bugs. (Of course if you don't test something very well, you don't know about the bugs, do you?)

The problem doesn't seem to be one of company size. Some of the smallest suppliers have a better track record than some of the bigger ones. One of these recently released a new version of their product that had undergone three months of in house testing. Yes, there were a few bugs in the release version, but the supplier had them all corrected within a couple of weeks, and an update mailed very quickly.

I have next to me a letter to a supplier half way around the world from Ann Arbor MI. He sent me a new version of his compiler last January with REAL variable capability for the first time. Very little of that capability worked properly, past the basic four functions. I sent half a dozen simple test programs that each illustrated a problem. Two months later I received another version with the problems I had reported all repaired. I went at a program using the capabilities of the compiler and I got about 10 lines farther than I did the first time and found another serious problem. Air mail takes two months round trip to this supplier. It is now the end of March. By the end of May, approximately, I can expect to receive another update for trial. If you were that supplier, and seriously interested in selling a product in the near future, wouldn't you try a little harder to test your product comprehensively? (Or as I have suggested, find a tester a little closer to home)

Now for a lecture to you suppliers... You MUST test every function of your product. If it is a compiler and it has long integer variables, multiplying 3 by 4 and getting 12 back is NOT a test of the long multiply!! Such multiply routines frequently show problems at the boundary between half and full length. Try 65535 * 32768 for a result that is nearly the maximum representable number. Try positive and negative numbers in all possible combinations. Try ZERO. If you have float, double, short, integer, and long, you have to try the logical comparisons for ALL of them. You can't argue "Well the integer comparisons worked and the long are patterned after them, so I thought I didn't have to test them." You say all these tests will cost you more than you can stand to pay out to your programmers, or delay cash income longer than you can afford? Then raise the price of your product so the return will be greater when you get the product on the market. Most of your customers would rather pay an additional $50 or $100 for a good piece of software, and know that they are not going to have to fool around for 6 or 8 months before all the bugs are out.

If you reduce the bugs to absolute minimum before shipping product, you will have minimal expense repairing the remaining few bugs, and better yet, you'll have instilled some confidence in your customers, who will be glad to purchase your next product.

Now, may I say something in defense of the software suppliers. (I am on that side of the fence too, with my JUST text processor). A few years ago, we microcomputer users were happy to have a line editor such as TSC EDIT, and elated when something as fancy as the text processor PR came along. Those are both very good software, and this is not intended to knock them in any way. They had few bugs when they were introduced, and they served us well for a long time. Look, however, at the relative complexity of some of the software about which I have been talking here. A Pascal or "C" compiler is certainly many times more complex than a line editor. While it might be relatively easy to test something in the way of "application software" such as a text processor, no supplier can envision what users are going to do with a compiler. The vast majority of them certainly try to test the product reasonably thoroughly, but perhaps they don't go as far as a quick run through Kernighen and Ritchie to develop a check list of what features should run in their compiler (read the equivalent documentation's name here for other compilers). Then next week someone will try a peculiar construct that is legal in the language, but perhaps little used, and some little detail will cause it to fail.

Some of the software suppliers have developed a list of users who tend to test the software thoroughly, perhaps only because of the applications for which they use it, and those suppliers frequently send out "pre-release" versions for "field tests". I can truthfully say that I have never received a piece of software for such a test and not found a bug or two nearly immediately. Of

course that is the idea of the test, and the suppliers gratefully supply the testers with upgrades and updates until the final product is released. I've found just about ALL the suppliers to be VERY much interested in removing all the bugs they can, and as quickly as possible. Think for a moment about the software supplier's position. The sooner he gets all the bugs out of his product, the less revisions he will have to mail out to his customers. Mailing new disks and revised manuals is a costly operation that can make the slim profits disappear and even turn them into a loss.

If you purchase new software and find a bug, have patience with the supplier for a reasonable amount of time. Perhaps 50 other users found other bugs and he is trying very hard to remove them all before sending out updates. If you find a supplier very unresponsive, both '68' Micro Journal and I would like to hear about it. Either we can help by getting the communications established better, or perhaps we can help directly, should the problem turn out to be one of misunderstanding the instruction manual or the purpose for which the software was intended.

This note is being appended a week later... At the time I wrote (and rewrote and rewrote) the above, I was very discouraged because of a long series of "buggy compilers" that have come to me all at once. I don't mean to sound "harpy" or to repeat myself endlessly on the subject of testing and correctness, so I sound like a broken record. I do think some of the suppliers don't try hard enough to test their wares before they sell them. Software projects nearly always take longer than estimated and a small company must run out of capital eventually, and have to get some product shipped to stay alive.

If anything ever kills the SS-50 bus, it will simply be the fact that the software market is too small and the suppliers can't make it on sales of their wares.

```
 NAM BCD#6T

PC      EQU 8
WARMS   EDU $CD03

PI      FCB $41,$31,$41,$59,00
RHO     FCB $41,$45,$28,00,00 DENSITY OF STEEL
HEIGHT  FCB $41,$68,$50,00,00 6.85 INCHES
RADIUS  FCB $41,$27,$50,00,00 2.75 INCHES
WEIGHT  FCB 0,0,0,0,0         PLACE FOR RESULT IN OUNCES

BEGIN   LDX #RADIUS
        LDY #TOTAL
        JSR FPMOV       MOVE RADIUS TO TOTAL
        LDX #RADIUS
        BSR MULMOV      SQUARE RADIUS
        LDX #HEIGHT
        BSR MULMOV      MULT BY HEIGHT
        LDX #RHO
        BSR MULMOV      MULT BY DENSITY
        LDX #PI
        BSR MULMOV      MULT BY PI
        LDY #WEIGHT
        JSR FPMOV       MOVE TOTAL TO WEIGHT
        JMP WARMS
```

```
    LIB FPMULBCD.SRC

    END BEGIN

 NAM FP8CD

PI      FCB 0,0,0,3,$14 PI
RHO     FCB 0,0,0,4,$53 DENSITY

HEIGHT  FCB 0,0,0,6,$85 INCHES
RADIUS  FCB 0,0,0,4,$35 INCHES
WEIGHT  FCB 0,0,0,0,0   OUNCES

BEGIN   LDX #RADIUS
        JSR MOVINT      MOVE RADIUS TO TOTAL
        LDX #RADIUS
        BSR MOVMUL      SQUARE RADIUS
        LDX #HEIGHT
        BSR MOVMUL      MULT BY HEIGHT
        LDX #RHO
        BSR MOVMUL      MULT BY DENSITY
        LDX #PI
        BSR MOVMUL      MULT BY PI
        LDX #WEIGHT
        JSR MOVTOT      MOVE TOTAL TO WEIGHT
        JMP WARMS

* SUBR TO MOVE [X] TO MULBUF AND MULTIPLY
* TOTAL BY MULBUF WITH RESULTS IN TOTAL

MOVMUL  JSR MOVINM      MOVE TO MULBUF
        JMP MULBCD      MULTIPLY

        END BEGIN
```

```
 TTL GETNUM
* SUBR TO INPUT A 10 DIGIT BCD NUMBER
* C/R = ERMINATION - COMMA IS IGNORED
* ONLY TWO DIGITS ALOWED AFTER DP
* ESC RETURN WITH V SET
* EXIT WITH X POINTING TO NUMBER, USES A & B

GETNUM LDX #NUMBER      POINT TO DESTINATION
       CLR DP DECIMAL   POINT SW = 256
       JSR CLEAR5       CLEAR NUMBER BUFFER
       LDX #NUMBER      POINT TO NUMBER
GNUM1  JSR INPUT        INPUT CHAR (NO ECHO)
       CMPA #$1B        (ESC) ABORT INPUT?
       BEQ GNUMX        YES,RETURN
       CMPA #$0D        TERMINATE ENTRY?
       BEQ GNUM4        YES, ADD DECIMAL DIGITS
       CMPA #$39        NUMERIC?
       BHI GNUM3        NO, RE-ENTER
       CMPA #$2C        NUMERIC? /, -, OR DP
       BMI GNUM3        NO, RE-ENTER
       JSR OUTT         YES, ECHO CHARACTER TO TERMINAL
       CMPA #$2C        <,>?
       BEQ GNUM1        IGNORE COMMA
       SUBA #$30        MAKE IT BINARY
       BPL GNUM2        NOT A DP
       LDA #2           ALLOW ONLY 2 MORE DIGITS
```

```
        STA DP          SET COUNT
        BRA GNUM1       CONTINUE
GNUM2   BSR TRUCK4      ROTATE NUMBER 4 BYTES LEFT
        ORA 4,X         ADD LSB TO BINARY INPUT
        STA 4,X         SAVE IT
        DEC DP          DONE?
        BNE GNUM1       NO, CONTINUE
GNUMX   RTS
GNUM3   LDX #RENTER     MESSAGE TO RE-ENTER NUMBER
        JSR PDATAS      PRINT MESSAGE
        BRA GETNUM
GNUM4   LDA DP          HOW MANY DECIMAL ZEROES TO ENTER
        CMPA #1         ONE?
        BEQ TRUCK4      YES
        BSR TRUCK4      ENTER TWO DECIMAL ZEROES

* SUBR TO ROTATE LEFT

TRUCK4  LDB #4          NUMBER OF TIMES TO ROTATE
TRUCK   ASL 4,X
        ROL 3,X
        ROL 2,X
        ROL 1,X
        ROL 0,X
        DECB
        BNE TRUCK
        RTS
```

OS9 USER NOTES

By: Peter Dibble
517 Goler House
Rochester, NY 14620

The OS-9 Seminar, OFlex, New Manuals
Some C Functions and Other Stuff

I went to the OS-9 users seminar last summer, so did almost every person I've heard of in the OS-9 community. It was interesting walking through the exhibit hall and listening to the speakers. The thing that makes me willing to go halfway across the country to take part in the seminar this summer is the fun I had last year talking with other OS-9 people. Most of us, myself included, spend our lives in a world where every other microcomputer user thinks the world ends right past PC-DOS and CPM. Last summer I fairly wallowed in the pleasure of being with hundreds of people who shared my interest in OS-9. We argued, agreed, complained, puzzled, and applauded about things that are dear to OS-9 users (and not many others).

If you need a practical reason to spend a long weekend in Des Moines, bring a question with you. If you have been itching to show the person on the Microware hotline a problem that he can't reproduce, he'll be there. Go demonstrate the problem yourself. If you want to suggest that OS-9 badly needs a WALL command you can probably find someone important and back him into a corner about it.

All the important vendors were there last year -- I assume they'll be back. If they came, you'll be able to check the Smoke Signal version of OS-9 for compatibility with other versions. Try a few things on the GIMIX III. I hope Privac comes again; their graphics board is much more impressive in motion than in an advertisement. I imagine there'll be a bunch of new vendors there showing CoCo products.

The vendors and Microware staff notwithstanding, the best place to look for answers will be standing or sitting beside you (very likely at breakfast or some other improbable time). Last year I found the other users at the Seminar a mine of useful information. If you are a vendor, go to the Seminar even if you don't have a booth. It is a great place to test the water.

The Seminar is a businesslike affair, but it is also something of a party: Jeanne Kaplan's party. Everyone who has dealt with Microware for any length of time knows that Jeanne is a consummate organizer. Last year everything ticked along smoothly despite the fact that she must have been slowed down a little by the child she was about to have. Last year Microware hosted a banquet and a fancy brunch. The Governor of Iowa came and gave us a little talk over dinner. Ken Kaplan handed out prizes to individuals who had made particularly distinguished contributions to the OS-9 community. At the brunch more prizes were handed out. I wonder what is in store for us this year.

Microware is going to give the Users Group some software for a raffle. I don't know just how it will be organized yet, but the plan is to hand the prizes out at the Sunday morning brunch.

Last year we heard a lot about the new 68000 version of OS-9. This year we may be able to see one in action. That's not official from Microware, but there are signs that it may be ready.

I guess it sounds like I'm advertising the Seminar. I suppose I am. I wouldn't miss it for the world, and I hope I'll see you there.

OFlex

Just today I received a copy of OFlex. This program runs Flex as a process in an OS-9 Level Two system. I'm afraid it's been too long since I used Flex with any regularity for me to give the program a good workout. Still, I ran a few Flex programs and checked out the interface to OS-9.

I remembered from "The Soul of a New Machine" that Adventure was an important test used on new hardware. I have a version of Adventure which runs under Flex, so I ran through a dozen rooms or so with it and grabbed two or three treasures ... no problem. I compiled a Pascal program using the TSC Pascal compiler with no difficulties except some trouble remembering how to use Flex.

Part of the OFlex package is a program called XCOPY that runs under OFlex. XCopy can copy from OS-9 files to Flex files and back. I tried every combination I could think of and couldn't make it fail. That brings up the one important failing I could find in OFlex; there is no FORMAT utility. I guess FORMAT is too near the hardware to run in what amounts to a virtual machine.

OFlex can read and write Flex disks. It can also format files on an OS-9 disk so the files can be treated as Flex disks by OFlex. The files are accessed through a command called ASNDISK. Using ASNDISK, files can be associated with each disk number (1 through 4). This is a useful feature for Flex. I shudder to think of the problem it would be dealing with a hard disk full of Flex files. With OFlex the hard disk can be broken up into many

smaller virtual disks giving manageable bunchs of files to work with.

OFlex isn't reentrant. This is sad, but, as I remember it, many Flex programs change flags and pointers inside Flex. Because it isn't reentrant, each instance of OFlex running under OS-9 needs a full 60K, but, if the memory is available, many users can run OFlex on the same machine. This could be viewed as an easy way of getting multi-user Flex.

OFlex is licensed from TSC and Frank Hogg Labs. As far as I can tell it is regular Flex with modified I/O which feeds into OS-9. It ran the programs I tried flawlessly, but I know of several Flex programs (I've written some myself) which use memory-mapped I/O directly instead of going through Flex. They won't work under OFlex. Anyhow, if you have OS-9 and you wish you could run most of your old Flex programs, or at least read the old disks, OFlex will do what you need. If you have no particular need for OS-9 but figure OFlex might be an improved way to run Flex, you must be very brave. It is an improvement over regular Flex in several ways, but one day a program you desperately want to run won't work with this mutation of Flex. In any case try OFlex with your software before you rely on it.

New Manuals

I got a stack of new OS-9 Manuals last week. I'm not an authority on most of the OS-9 Manuals, but I've practically memorized the System Programmer's Manual. The new manual is a big improvement over the old one. There is a section on memory management for Level Two and a section on pipes with a few assembly language examples. The Level Two Service Requests are in with the other requests, not isolated in an appendix. Speaking of Service Requests, the manual goes into a good deal more detail than it used to on some of them. The explanation of Chain takes more than two pages, Exit takes about a page and a quarter, as does Intercept.

The new manual contains lots of useful snippets of code demonstrating tricky points. I was particularly pleased to see five chunks of about ten lines each that cover the most obscure parts of an interrupt driven device driver. I believe those chunks of code were taken straight out of the ACIA device driver.

Microware has been producing steadily better manuals for the last two years. The new Systems manual is their best so far. If it had been available last January, I might never have seen a need for this column.

C Functions

I have been working on a program to model a problem in distributed systems for a course I am taking. I needed some functions to manipulate floating point numbers as a separate mantissa and exponent. I spent most of an evening fussing around with assembler before I gave up and wrote the functions mostly in C. It was such a frustrating experience that I decided to include them in this column. I wrote frexp and modf to duplicate functions that are part of the UNIX math library.

Frexp returns the mantissa of val as a double less than one, and stores the exponent in the integer pointed to by eptr. The exponent is for a power of two; that is, the number was (val=x*2**exp).

Modf separates a double into an integer part and a fractional part. The integer part is stored

Plan to Attend the 3rd ANNUAL OS-9 USERS SEMINAR

August 17, 18, 19, 20
Pre-Registration Only!

- MORE INFORMATION
- MORE EXHIBITS
- MORE SPEAKERS
- HARDWARE
- SOFTWARE

- TECHNICAL SESSIONS FOR 6809 & 68K

Plan now to attend the 3rd Annual OS-9 User Seminar. This is an event you won't want to miss if you use, sell or are interested in systems that use Microware 6809/68000 software. Informative round-table discussions on almost every aspect of the design and use of Microware software will be held. A bigger and better exhibit area will have display booths from many of the leading suppliers of OS-9 compatible hardware and software. Don't miss this chance to increase your knowledge and skill in the latest microcomputer software technology — Register today!!!

Fee: $125

Location: Marriott Hotel
Des Moines, Iowa

Don't Miss It: Pre-Register Now!
Call: 515/279-8844 or Write:

MICROWARE SYSTEMS CORPORATION
P.O. Box 4865 • Des Moines IA 50304

OS-9 and BASIC09 are trademarks of Microware and Motorola.

at the address in ptr (as a double), and the fractional part is returned (also as a double).

I wrote most of the code for these functions in C because I couldn't do it in assembler. I certainly tried, but Microware C uses lots of internal subroutines and a special static storage location called flacc (floating point accumulator) to do floating point calculations. I had lots of trouble finding the floating point number and returning the number to the caller. As you can see from the programs, my solution was to use C to do everything in modf, and to find val and return a value in frexp.

The Butterfly

It looks like the Computer Science Department here at the University of Rochester is going to get a computer called a Butterfly. It is named after the network used to connect its processors together. The Butterfly that will be coming here has 128 68000 microprocessors. Each 68000 has at least 512K of memory and, potentially, its own buss. They are all able to read and write one another's memory. I hear that this computer will have the fastest instruction rate in the world. Of course, instruction rates are an almost meaningless measure, but won't that be a marvelous computer to develope parallel algorithms on! It's coming with a UNIX-like operating system, but I can't help but wonder whether it could run OS-9.

Dynaspell

Last summer at the OS-9 Users Seminar I met Dale Puckett at dinner -- before we were both elected as Users Group officers. I had been a loyal user of Dynaspell, a program written by Dale Puckett, but I wasn't entirely happy with it. In fact I had written a very mixed short review of it in this column. During dinner I made Dale sit through a careful explanation of my criticism of his program, and a long discussion of what I thought a spelling checker should do.

Dale was very patient with me. He even encouraged me to go into more depth about my ideas for the perfect spelling checker. I told him that I would write a new, more complete review of Dynaspell if he would send me a version that deserved fresh considaration. Some months later I got a package from Dale including something pretty close to my dream spelling checker. We went through some iterations working out various problems. Now I owe Dynaspell a review. I have been very slow about writing that review, so let me summarize here. I'll go into more depth another month. Dynaspell isn't perfect, but I haven't been able to find any bugs in the latest version. It is much faster than the early version I had. It is able to look near misses up in its dictionary and suggest corrections when it suspects a spelling error.

My remaining complaint about Dynaspell is that the new features don't go far enough. The "look up" feature isn't as selective as a would like. It often finds more possible spellings for a word than it can fit on the screen. On the other hand it sometimes doesn't search widely enough to find the correct spelling for me. I also wish it would give me the features of a screen oriented text editor when it finds a spelling editor. Dynaspell has a mode in which spelling errors can be viewed in context, but the context it shows is a screen-full of the document up to and including the word in error. I would like to be able to move forward and backward throught the document, and to change words other than the one in error.

I used my early copy of Dynaspell because I need a spelling checked badly and it was the best I had. I use it more often and more happily now. It is one of the best spelling checkers I know: mainframe programs included.

A Nice Experience

Early last summer I bought a TeleVideo 970 terminal. They were just becoming available on the market; in fact, I had a hard time finding one. It seems the boat bringing a large shipment in from overseas had sunk. I'm not certain I believe that, but it was definitely difficult to find one to buy. I finally found one, got it home, and started using it. Nice terminal. Big screen, nice keyboard. Almost too flexible.

After about a week I started finding bugs. A few commands didn't work right. I called the number in the manual and talked to an engineer. The next day I got a package via Federal Express with new firmware ROMS. That wasn't the end of the problems with the terminal. I'm one of those annoying people who reads the entire manual then tries all the strange combinations of commends just to see what they will do, and the 970 has a manual about two thirds of an inch thick. The last time I called them I told them that I needed a feature which was documented in the manual, but which the errata with the manual said was not implemented (downloadable fonts). Without a complaint they sent me a whole new logic board which supports that feature.

I don't think I would recommend the TeleVideo terminal to most OS-9 users. The terminal costs over a thousand dollars. That makes it hard to justify when a adequate terminal only costs five or six hundred dollars. For those who take terminals seriously, it is worth what it costs. It supports ANSI standard and VT52 control sequences, and includes about every feature I can imagine except full graphics (they say that's coming).

The best thing about the 970 is the excellent support TeleVideo gives. Many large vendors seem to lose interest after they sell you their product. TeleVideo has gone out of their way for me again and again.

Tricks for Level Two

I just learned about OS9P3 in the new OS-9 System Programmer's Manual. I have often wished for an easy way to add System Service Requests to OS-9. Under Level One, it isn't too hard, but under Level Two it has required either slight of hand or very strange practices. Only modules running in the system address space can add Service Requests, but OS-9 doesn't include a way to run a process in the system address space. I have run device drivers and file managers just to add Service requests, and considered renaming OS9P2 as OS9P21 and adding my own OS9P2 which will link to and call OS9P21.

Microware has included something like that last trick in Level Two. After OS9P2 is finished initializing (all it does is set up a list of Service Requests) it tries to find OS9P3. There is no OS9P3 unless the user adds it to the boot file, so it generally fails to find the module, but if it finds OS9P3 it executes it as a system module. This opens up lots of interesting possibilities.

Other interesting possibilities are suggested by the SS.SIG and SS.Relea SetStat codes. SS.SIG instructs OS-9 to send a specified signal when data is ready from a path. The easy use for this is to wait for output from several paths at once. This is especially good for things like "modem" programs

that need to wait for input from two paths simultaneously. Without this SetStat the only way to handle that problem was to poll both paths.

It isn't difficult to write a program that polls a number of paths. In fact, polling is the way most of the more primitive microcomputer operating systems work. The pro lem with polling is that it wastes tremendous amounts of CPU power. I seldom type faster than 2 characters per second. If a program has to poll for my input it will look for something to read thousands of times efore it gets anything.

With SS.SIG it should be possible to do e couple of SetStats and wait for a signal. While an OS-9 program waits it uses essentially nothing but memory. This should make modem programs and other programs with similar pro lems much more efficient.

The other use I can think of for SS.SIG is to solve the problem that devices can't be preempted. If you have a system with more than one terminal you have probably noticed that if you send a message to another terminal, the message waits until the user at the other terminal types a carriage return. That's because there is a program (e.g. the shell) trying to read from that terminal. Until the read is finished OS-9 won't allow any process to write to it. SS.SIG gives us a way to break that deadlock by not leaving a read active.

I have included a trivial program which demonstrates the use of the SS.SIG setstat with this Column. It doesn't do anything useful -- just copies lines from standard input to standard output. The exciting thing is that it works! I ran tstssig on one terminal; typed a few lines into it to make certain that it worked; left it at its prompt, and went to my other terminal. I typed

```
Echo Hi there >/term
```

on the other terminal and it appeared immediately on the terminal running tstssig. I went back to the terminal running tstssig and typed a blank. The blank caused a signal to be sent to tstssig letting it proceed to the I$ReadLn. Once the read was "up" /term was locked. I tried to send another message to /term and found that I had to wait until I typed a carriage return on /term before the message was delivered and the echo command completed.

I wonder whether the SS.SIG trick should be used as a matter of policy when long waits for input are expected.

Editor's Note: As this issue was being prepared for final press, clarification of the Smoke Signal Broadcasting versions of OS-9, were received from SSB.

The net result of everything that has been discussed concerning this over the past few weeks, is simply this:

"Any SSB customer or user who is presently using the 'enhanced' version of OS-9 1.2 from SSB, or who contemplates using OS-9 on a SSB computer, can actually have either/or both." All that is required is that the user notify SSB and the proper EPROM for either, along with either Microware's (virgin) OS-9 1.2 or SSB enhanced OS-9 1.2 will be furnished, at no extra charge, so we are informed by Jim Allday of SSB.

The net result is that SSB users have a choice. again, we see responses from SS50 suppliers that far exceeds the cooperative spirit found with other groups of micro users, and something I, for one, am proud of.

DMW

- - -

```
00001                           nam   tstssig
00002                           ttl   Test SSIG set stat
00003          *------------------------------------------------------*
00004          *        Test SS.SIG SetStat Service request.          *
00005          *   This program will copy lines from standard input   *
00006          *   to standard output without tying the device        *
00007          *   used for standard input up with a read, or using   *
00008          *   excessive amounts of CPU time by polling the       *
00009          *   standard input path.                               *
00010          *                                                      *
00011          *   tstssig has no practical use that I can think of.  *
00012          *------------------------------------------------------*
00013                           IFP1                use os9defs
00015                           ENDC
00016  0011     Type    set   Objct+Prgrm
00017  0081     Revs    set   ReEnt+1
00018  0001     StdOut  set   1
00019  0000     StdIn   set   0
00020  0000     SSCode  set   0          code used to indicate input wa
00021  0064     LineSiz set   100
00022  00C8     Stacksiz set  200
00023  0000 87CD0072           mod   TstLen,TstNam,Type,Revs,Entry,MemSize
00024  0008 54737473 TstNam    fcs   /tstssig/
00025  0014 01       Edition  fcb   1
00026  0015 303D8E   Prompt   fcs   /0=>/
00027  0003          PromptL  equ   *-Prompt
00028          *------------------------------------------------------*
00029          *                Static Storage                        *
00030          *------------------------------------------------------*
00031  0000     IntNo   rmb   1          Save the signal from the trap
00032  0001     Line    rmb   LineSiz    Storage for a line to echo
00033  0065             rmb   Stacksiz
00034  012B     MemSize equ   .
00035  0018     Entry
00036          ********
00037          * Set up signal intercept trap
00038          *
00039  0018 308D0050           leax  Trap,PCR    Address of interrupt trap code
00040  001C 103F09             OS9   F$Icpt
00041  001F           Loop
00042  001F 308DFFF2           leax  Prompt,PCR
00043  0023 108E0003           ldy   #PromptL
00044  0027 8601               lda   #StdOut
00045  0029 103F8A             OS9   I$Write     Write the prompt
00046  002C 2536               bcs   Error
00047  002E           StrtRead
00048  002E 8600               lda   #StdIn
00049  0030 C601               ldb   #SS.Ready
00050  0032 103F8D             OS9   I$GetStt    any data ready?
00051  0035 2516               bcs   DoSSIG      No; wait for a signal
00052  0037           DoEcho
00053  0037 3001               leax  Line,U
00054  0039 108E0064           ldy   #LineSiz
00055  003D 8600               lda   #StdIn
00056  003F 103F8B             OS9   I$ReadLn    Read a line
00057  0042 2520               bcs   Error
00058  0044 8601               lda   #StdOut
00059  0046 103F8C             OS9   I$WritLn    and echo it back out
00060  0049 2519               bcs   Error
00061  004B 2092               bra   Loop        Go prompt for the next line


00062  004D           DoSSIG
00063  004D C61A               ldb   #SS.SSIG    setstat function code
00064  004F 8E0000             ldx   #SSCode
00065  0052 103F8E             OS9   I$SetStt
00066  0055 8E0000             ldx   #0
00067  0058 103F0A             OS9   F$Sleep     Sleep until an interrupt comes
00068  005B B600               lda   IntNo
00069  005D C104               cmpb  #SSCode
00070  005F 27CB               beq   StrtRead
00071  0061 43                 coma              set carry
00072  0062 2000               bra   Error
00073  0064           Error
00074  0064 C103               cmpb  #E$Eof
00075  0066 2601               bne   Exit
00076  0068 5F                 clrb              EOF isn't an error
00077  0069           Exit
00078  0069 103F06             OS9   F$Exit
```

```
00079          ****************
00080          *  Trivial interrupt trap
00081          *
00082  006C            Trap
00083  006C E7C4               stb   IntNo,U    save the interrupt code
00084  006E 3B                 rti
00085  006F BA1340             swod
00086  0072            TstLen  equ   *

00000 error(s)
00000 warning(s)
$0072 00114 program bytes generated
00120 00301 data bytes allocated
$2221 08737 bytes used for symbols
```

```
 1 $double
 2 frexp(val,iptr)
 3 double val;
 4 int *iptr;
 5 {
 6     register double rp;
 7     int exp;
 8
 9     rp = bval;
10     /* at this point U contains the address of val */
11 $asm
12   ldb 7,U get C exponent
13   addb #128
14   sex
15   std ,S save exp
16   lda #128
17   sta 7,U
18 $endasm
19
20   *iptr = exp;
21   return(val);
22 }
```

```
 1 /* modf returns the positive fractional part of val.
 2    and stores the integer part in the double pointed to
 3    by ptr.
 4 */
 5 #define MAXLONG 134217727
 6
 7 double
 8 modf(val,ptr)
 9   double val, *ptr;
10
11 {
12     double tmp;
13
14     if(val > MAXLONG)
15        {
16           *ptr = val;
17           return(0.0);
18        }
19
20     tmp = (long)val; /* truncate to int by coercion to long*/
21     *ptr = val - tmp;
22     return(tmp);
23 }
```

RUMORS & SUCH

Rumors and Such

As you might recall, I occasionally dump some fact and a little fiction here under this heading. Been pretty good so far - 90% or more! Now that is better than most of the big boys do. But then I have better sources than they do. That and a lot of reading and listening between the lines gets it done.

Since my bout with the pump doctor I have been pretty far removed from the normal stream of things around here, but I now am back in the swing, more and more each day. So as is my usual habit I looked around for something to 'rumor' about. Really not much as actual rumor except that Tandy may yet come out with a much improved color (6809) computer.

Seems that they DO NOT want to merchandise anything to compete with their Intel CPU driven computers, on a serious level. Fact is, they are surprised at what has happened to their 'little toy' color computer. I can remember telephone conversations with Tandy personnel, some years ago when they first came out with the CoCo. Our first one had a serial number less than 200. I was telling them what a great foundation the box made and they were touting their 'lousy' chess game. They didn't even consider disks for it at that time. Boy, what a difference disks and OS-9 has made to the little machine. A true 64K, multi-tasking, multi-user (both difficult due to built in hardware designs), but there still. I guess there are over 30,000 OS-9 users now on the CoCo and more coming each day, just goes to show what a fine CPU the 6809 is. A shame Motorola never felt as strong as we have about the 6800/6809. If they had given the 6800 and the 6809 as much attention and exposure as Intel did the 80 series of their devices, I guess all the IBM's and clones would be 6809 based. The old 'mouse trap' saying still holds true. The 68000 has received better, but that is another ball game of sorts.

A New 6809 Computer from SWTPC

For the past several months a new computer from SWTPC has been advertised in 68 Micro Journal. We have received a lot of telephone calls from prospective users wanting some specs. Well we have some pretty solid info concerning the new SWTPC X-12+. Here goes:

The X-12+ system is both a FLEX™ and UniFLEX™ system. THe heart of the system is the 6809 CPU supported by three (3) additional CPUs.

First, there is a 3870 Keyboard enco r, which is actually a complete CPU and unburdens the system from keyboard chores. This unit is interfaced to the main system by a separate 6809 terminal controller. Which leaves the main CPU with little to do ab t terminal or keyboard activities.

Of course as stated above the main CPU is a 6809. Faster than a 68000 in many functions, except for math operations, well - here SWTPC has brought in a new player (option), the TI 320 32 bit, bipolar, 20 Mhz Floating Point processor. Talk ab t fast!

The system comes standard with 256K RAM and is expandable to a full 1 megabyte of RAM.

A FLEX version comes with floppies and a special version of FLEX that will allow a single user operation of multi-tasking capabilities. This is accomplished by dividing the RAM into two units. Each RAM unit will allow 64K for a task and 64K for 'super speed' disk type RAM. Under this scheme a user will run two separate tasks with very fast data accessing, provided the data is moved to RAM-disk. This is similar to the foreground/background facility of the "DYNASHARE' FLEX program sold by S.E. MEDIA, and really makes a single user system fly!

In the FLEX mode the system is restricted to one user and two tasks, at super speed.

The (rumored) price of the 256K, FLEX single user system with floppies is in the neighborhood of $4500.00, less the math option.

The system is also available with UniFLEX and 20 megabyte winchester storage. UniFLEX can be run with floppies only but the winchester enhances UniFLEX operations to the point that any serious user will opt for the winchester version.

Under UniFLEX the system is expandable to three (3) users merely by plugging in additional CRT terminals (2) maximum.

The UniFLEX system with one floppy and 20 megabyte winchester will sell for approximately $6500.

The basic system comes with a parallel port and 4 serial ports, two of which are designated for terminal connection.

I am sure that a call or letter will get you more information from SWTPC.

SSB Enters 68000 Arena

Smoke Signal Broadcasting in this issue announces their new 68000 computer systems (see Bit Bucket - SSB product announcement). The press release is fairly complete as concerns the specs of their new VAR/68K

systems.

However, there is a possibility of a price break for 68 MICRO JOURNAL readers, that I have to tell you about. It is rumored that for the month of JULY 1984, 68 MICRO JOURNAL readers will have the opportunity to purchase a VAR/68K 68000 SSB computer for something like **50% OFF THE REGULAR ADVE TISED PRICE!!!!**

Now let me tell you that is one sweet deal! Thats right a total 68000 system with terminal, floppy, winchester and 256K RAM with a UNIX™ type operating system for **half price!** Come by our office after July and see our new VAR/68K 68000 - I guarantee you we are going to get one.

For space considerations I cannot repeat the news release particulars but if you want to know more, I just bet that the folks at SSB will certainly be more than happy to reply to your call or letter, concerning this or any of the other fine SSB products.

Oh, I believe that they will also make available to any Standard S50 Bus user a full set of boards, at a very reasonable price, to allow you to update your present 50 pin computer to equal the VAR/68K 68000 SSB computer. Another savings for those considering updating to the 68000/68008. Watch coming SSB ads and 68 MICRO JOURNAL announcements concerning the availability and price of the 68000 system boards.

**

Thanks Smoke for making our readers such a fine offer!

DMW

- - -

** Late Note: Just received a call from SSB concerning the boards above, can't believe what they implied, but I'm going to tell you what I think they told me:

For those with a Standard S50 Bus computer the following parts are packaged up into a kit for 68 MICRO JOURNAL readers:

1. A new heavy duty motherboard.
2. A new standard heavy duty power supply. (handles every thing including winchester)
3. New SSB 68008 CPU board.
4. A new SSB 5-8 inch disk I/O controller.
5. A new SSB winchester disk controller.
6. A new 5 megabyte winchester disk drive.

Thats right, **everything but I/O and memory for**
Get this - appx. $2500. RIGHT $2500.

This is, to the best I can figure about $7,000.00 worth of boards and winchester drive. A real bargain for sure and for those wanting to upgrade - **HERE IT IS!!!!**

THANKS AGAIN SMOKE!

- - -

"C" User Notes

Edgar M. (Bud) Pass, Ph.D.
1454 Latta Lane
Conyers, GA 30207

INTRODUCTION

I have assumed the responsibility of writing the C User Notes column from Norm Commo. His articles were interesting and one of the features in '68' Micro Journal which I always read. Some of you may remember that I wrote a series of articles for System 68 magazine, which has since ceased publication.

CALL FOR ASSISTANCE

I definitely need help from '68' Micro readers and vendors to supply me with a continuing supply of interesting C-related programs, functions, compilers, application programs, books, commentary, etc. to review and to discuss.

I also would like to publish one reasonably short C program or function each month, preferably with explanation. User input for such a concept is vital.

I need submissions on disk. I can accept standard FLEX, OS/9, UNIFLEX, COCO, CP/M, IBM PC, and a few other standard formats. I cannot accept non-standard formats. To protect '68' Micro and myself, I need a written release from the owner of the program or function.

If you have questions about, problems with, or success with a particular C-related product, send me a letter and/or disk (with appropriate documentation) to the address above, not to the '68' Micro address. I will accumulate them, comment on them, and publish the more interesting ones, but I will attempt to answer all of them. Maybe your experiences and questions are not as unique as you might think. I know that each of the C compilers available for the 6809 has its own set of strengths and weaknesses. Maybe publicity can help get the vendors to correct their problems.

C COMPILER 6809 IMPLEMENTATIONS

There are several implementations of C on the 6809. Like most other products, the completeness and usability of the implementations vary considerably. However, there are several major groups into which many of the implementations fall. The most complete compilers are called "Full-C" and implement UNIX C almost entirely. The least complete compilers are advertised as "Small-C" and implement only specific subsets of UNIX C. Many C compliers are complete in some areas, such as verbs, but are incomplete in some areas, such as floating-point variable types. Many C compilers claim to be complete implementations of UNIX C, but actually provide far fewer facilities.

The following list presents most of the major known versions of C for the 6809, loosely ranked in decreasing order of completeness of implementation:

Microware/McCosh	Full C
Introl	Full C
Telecon	Full C
Wordsworth (v2)	Small C
Dugger (v2)	Small C
Dyna-C	Small C
Everhart	Small C
Wordsworth (v1)	Small C
Dugger (v1)	Small C
Intersoft	Small C

Only Microware/McCosh, Introl, and Telecon currently claim to support floating-point, long variables, structures, unions, and initializers. None currently claim to support bit fields.

Essentially all of the non-Full C compilers were based on or derived from Ron Cain's Small C, which was written for the 8080. It has the advantage of being written in its own language, thus facilitating porting to new implementations. It has the disadvantage of having several major bugs, most of which were also ported to the new implementations.

The Telecon and Dugger compliers are apparently still marketed, but are not advertised in '68' Micro. The Wordsworth and Intersoft compliers are apparently no longer marketed. The Everhart compiler is in the public domain.

PUBLIC-DOMAIN C SOFTWARE FOR 6809

The Atlanta Computer Society Motorola Special Interest Group is attempting to establish a public-domain C library and bulletin board with upload/download capabilities. We have already converted a few programs from other public domain C libraries, but we need your help in making the library useful. I will print more information in future columns, including contents, addresses, phone number, etc.

USER COMMENTARY

Sidney Thompson has worked with the C language for several years, initially on UNIX systems, but then on micros. Several months ago, he discussed with me some ideas for an article based upon his experiences in attempting to use the Microware C compiler. I encouraged him to write the article and helped him rephrase and rewrite the notes. I have extracted some of his major points and have included them below.

> I have read several glowing reviews of the Microware C compiler. This compiler, along with FLEX and UNIFLEX versions, are the product of James McCosh, and thus share many of the same features and problems.
>
> I had been looking forward to using the compiler, since the McCosh compilers were the first 6809 C compilers to support structure initializers, as well as most of the other C language standards except bit fields.
>
> I decided that I should take a program that was already running under the OS/9 INTROL compiler and see if it would compile. This program consists of seven program modules plus a common "include" file that defines all the global variables and constants being used within the program.
>
> There were some trivial problems, such as TRUE and FALSE not being defined in the "stdio.h" file, but these were easily fixed.
>
> After the compiler had processed through the seven modules and started to link, I got an undocumented error message indicating a fatal "name clash", that all my variables had been previously defined.
>
> I determined that the problem was the declaration of the variables in the common "include file". The compiler was unhappy with either having all the variables declared as int and char, or having them all declared as extern int and extern char.
>
> The include file used in the module containing main() must NOT declare the variables as "extern" while all other modules MUST declare them as "extern". This means that the programmer must either use separate "include files" or must code the following alternate forms:
>
> ```
> /* before include file in main() module */
> #define GLOBAL
> ```
>
> or
>
> ```
> /* before include file all others */
> #define GLOBAL extern
> ```
>
> This is not documented in the Microware C compiler manual as a requirement to compile modules separately. This means that any source program that uses modules with a common declaration file and runs under UNIX, BDS, INTROL, or any of several other C compilers must be modified before it will work with the Microware C Compiler.
>
> There is no facility provided to modify the standard library, since the manual says that few people would need this capability. Yet, there are several missing key functions, such as the following:
>
> ```
> rand()
> realloc() (malloc() is provided)
> max()
> min()
> ```
>
> There are several, such as getime(), that differ from the equivalent UNIX function by having the function and its variables with different names.
>
> Like UNIX, Microware C requires the inclusion of header files for special functions, such as getime(), setstat(), isalpha(), pfinit(), etc. Yet the compiler does not always tell you if you do not include these files at the proper place, and the modules will compile and link. But they will not run.
>
> I was forced to remove almost all of the extra OS/9 level 1 modules from my os9boot file before the compiler could be run at all. With a standard 56k system, there is barely enough memory to use the C compiler. I understand that COCO OS/9 has more memory available.
>
> The preceding has attempted to present some constructive criticisms regarding the Microware C compiler. The following actions are recommended to help improve the usability of the product:
>
> A) Place the OS/9 level 1 version of the compiler into three or four passes, in order to reduce its memory requirements, and thus allow compilation on a normal OS/9 level 1 system, not a stripped-down one.
>
> B) Fix or document the problems associated with compiling C programs that use separately-compiled modules.
>
> C) Document all error messages, explaining possible causes and fixes.
>
> D) Add many of the missing functions to the standard libraries, and provide a means for a user to enhance, customize, or modify the libraries.
>
> With these changes, the Microware C compiler would be an even more impressive product than it already is today. This would lead to some serious business type application program development, rather than it being useful, in general, only as a tool for small programs.

I hope Sidney's entire article will be printed soon in '68' Micro. He is working on a follow-up article, which I am sure will be just as interesting as this one is. Whether you agree with him or not, he has some useful comments.

EXAMPLE C PROGRAM

Following is this month's example C program; it paginates listings.

```
#include "stdio.h"
main(argc,argv)
int argc;
char **argv;
{
    char *input,*output,*date;
    char line[256];
    int lines,pages,i;
    lines=99;
    pages=0;
    input=stdin;
    output=stdout;
    date=0xcc0e;   /* FLEX date locations */
    if(argc>1)
    {
        if((input=fopen(*++argv,"r"))==NULL)
        {
            fputs("can't open input\n",stderr);
            return(1);
        }
    }
    while(fgets(line,256,input)!=NULL)
    {
        if(++lines>58)
        {
            if(lines<99)
                putc('\f',output);
            lines=0;
            printf("    Page %d      ",++pages);
            printf("Date %d/%d/%d",
                (i=(*(date))&15),
                (i=(*(date+1))&31),
                ii=(*(date+2))&127));
            if(argc>1)
                printf("          File %s",*argv);
            putc('\n',outout);
            putc('\n',output);
        }
        fputs(line,output);
    }
    if(lines!=99)
        putc('\f',output);
    return(0);
}
```

Editor's Note:

I would like to take this space to first thank Norm Commo for the many, many hours of time he has devoted to the 'C User Notes. Norm was the first 'C' author and only full-time 'C' author we have had, until now. To say that Norm has done a good job, would be putting it too mildly. Norm has given more than many will ever realize. I want to THANK Norm, and his family who also had their time schedule intruded on by the constrains of getting a column out nearly every month. Without fine folks like Norm Commo, 68 MICRO JOURNAL would never have survived. I wish I knew stronger words, at times like these, but this is from the heart Norm, I thank you for all you have done and I thank your family for their understanding and consideration, for all those hours you gave to all of us. We are all enriched in this activity by your skill, effort and time. And it is our sincere prayer and hope that full health might be restored within your family, and soon.

THANKS Norm from all of us! GOD BLESS!

Most of you readers will recognize Dr. Bud Pass, who has authored many technical articles for various computer magazines, including System 68 and 68 MICRO JOURNAL.

I feel fortunate that Bud has agreed to take over the duties of getting out a 'C' column for all of us who strive to learn this new and exciting language. His qualifications are well accepted to carry on in Norms place. And he was the first choice of Norm to carry on.

So, "Welcome Aboard", Bud. I and thousands of others are looking forward to your leading us on into the depths and power of 'C'. And to all you readers I say, "enjoy."

DMW

- - -

READING NON-FLEX BASED DISKS

CONTINUED FROM LAST MONTH

```
2012 AE  E9 0001        LDX   TMP3,S    set current trk/sec
2016 86  01             LDA   #1        ADD means getnext trk/sec
2018 8D  0F             BSR   MAPPER    get next trk/sec
201A 6A  E9 0000        DEC   TMP1,S    gotten all sectors?
201E 26  D7             BNE   GB1       branch if not

2020 32  E9.0003        LEAS  LS6,S     release local storage

2024 1F  21             TFR   Y,X       copy ptr to end of buf to X
2026 35  26             PULS  Y,D       restore regs
2028 39                 RTS             return
                *
                * Data area for GETBLK routine
                *
          2029  LPC     SET   *
0000                    ORG   $0000
          0000  LOL     SET   *
0000            TMP1    RMB   1
0001            TMP3    RMB   2
          0003  LS6     EQU   *-LOL     len of local storage

2029                    ORG   LPC       restore PC


                *
                * Name     - MAPPER
                * Function - This routine takes in one of two kinds
                *           of parms. If ACC A is clear, then the
                *           contents of the X register will be
                *           taken as a CP/M block number, in which
                *           case the corresponding track/sector
                *           value will be returned in X. If ACC A
                *           is non-zero, then X will be taken as
                *           a track/sector value. In this case,
                *           the next track/sector value will be
                *           returned in X.
                *
                *           ACCs A and B are preserved
                *
          2029  MAPPER  EQU   *
2029 34  06            PSHS  D         save regs
2028 4D                 TSTA
202C 27  0F             BEQ   BLOCK     take as block value

                *
                * simply find next trk/sec value from map table
                *
```

```
202E 1F 10          TFR   X,D
2030 F6 207A        LDB   TBLOFF      get previous offset
2033 5C             INCB              go to next one in tble
2034 F1 0889        CMPB  MAPLEN      wrap around?
2037 25 34          BLO   GETIT       now, then ok

2039 4C             INCA              yes, incr track and
203A 5F             CLRB              offset to tble beginning
203B 20 30          BRA   GETIT       get sec and return
```

```
                    *
                    * convert block# in X to trk/sec equivalent
                    *
                    * The formula for performing the conversion
                    * from block # into 2 byte trk/sector value
                    * is as follows:
                    *
                    *     trk = (block# * 8) / TBLEN + OFFSET
                    *     sec = MAPTBL[ (mod(block# * 8,TBLEN) ]
                    *
          2030      BLOCK EQU   *
2030 F6 0889        LDB   MAPLEN      divisor
2040 F7 207C        STB   DIVISR      save it
2043 C6 08          LDB   #8          counter for divide
2045 F7 2078        STB   CTR         save it

2048 1F 10          TFR   X,D
204A 58             ASLB              dividend = dividend*8
204B 49             ROLA
204C 58             ASLB
204D 49             ROLA
204E 58             ASLB
204F 49             ROLA

          2050      DIVIDE EQU  *
2050 58             ASLB              shift dividend, quotient
2051 49             ROLA
2052 B1 207C        CMPA  DIVISR      is trial sub. successful?
2055 25 04          BCS   CHKCNT
2057 B0 207C        SUBA  DIVISR      yes, sub and set bit in quot.
205A 5C             INCB
          2058      CHKCNT EQU  *
205B 7A 2078        DEC   CTR
205E 26 F0          BNE   DIVIDE

2060 7D 087E        TST   @CONFIG     5" disk?
2063 27 04          BEQ   SET5        branch if so

2065 CB 02          ADDB  #OFFB       add in 8" offset
2067 20 02          BRA   SETDON
          2069      SET5  EQU   *
2069 CB 03          ADDB  #OFF5       add in 5" offset
          2068      SETDON EQU  *

                    *
                    * Acc A = remainder, Acc B = quotient
                    *
                    * Swap registers to set quotient,remainder pair
                    * and then use mapping table to find actual
                    * sector on CP/M disk
                    *
206B 1E 89          EXG   A,B         swap A,B

          2060      GETIT EQU   *
2060 F7 207A        STB   TBLOFF      save table offset
2070 8E 0887        LDX   #MAPADR     X -> mapping table
2073 E6 85          LDB   B,X         get sector
2075 1F 01          TFR   D,X         X has trk/sec value
2077 35 06          PULS  D           restore regs
2079 39             RTS               return
                    *
                    * Data Area
                    *
207A 00             TBLOFF FCB  0     hold current table offset
207B                CTR   RMB   1
207C                DIVISR RMB  1

          207D      MAPTBB EQU  *     Mapping Table (8")
207D 01 07 00 13    FCB   $01,$07,$00,$13,$19,$05,$0B,$11
2085 17 03 09 0F    FCB   $17,$03,$09,$0F,$15,$02,$08,$0E
```

```
2080 14 1A 06 0C    FCB   $14,$1A,$06,$0C,$12,$18,$04,$0A
2095 10 16          FCB   $10,$16
          001A      TBLEN8 EQU  *-MAPTB8

          2097      MAPTB5 EQU  *     Mapping Table (5")
2097 01 03 05 07    FCB   $01,$03,$05,$07,$09
209C 02 04 06 08    FCB   $02,$04,$06,$08,$0A
          000A      TBLEN5 EQU  *-MAPTB5
```

```
                    *
                    * Name     - READSS
                    * Function - This routine reads in one track/sector of
                    *            the disk in the drive specified by an FCB
                    *            pointed to by X. Reg D specifies the
                    *            track/sector to read in.
                    *
                    *            A BNE or BEQ should be used after calling
                    *            this routine to check for possible read
                    *            errors.
                    *
                    *            All registers except A are preserved.
                    *
          20A1      READSS EQU  *
20A1 ED 88 1E       STD   FCBCP,X     set trk/sec
20A4 86 09          LDA   #IRSS       set function code
20A6 A7 84          STA   FBBFC,X     set code in FCB
20A8 7E D406        JMP   FMSCAL      read and return
```

```
                    *
                    * Name     - MVC
                    * Function - This routine moves the "from" string
                    *            to the "to" string as follows:
                    *
                    *              X -> "from" field
                    *              Y -> "to" field
                    *              D = # characters to move
                    *
                    *            Regs A,B,X,Y are preserved
                    *
          20AB      MVC   EQU   *
20AB 34 36          PSHS  A,B,X,Y     save regs

          20AD      MOVE  EQU   *
20AD 34 06          PSHS  D           save len to move
20AF E6 80          LDB   0,X+        get a "from" char
20B1 E7 A0          STB   0,Y+        trans to "to" field
20B3 35 06          PULS  D           get len
20B5 83 0001        SUBD  #1          decr by 1
20B8 1083 0000      CMPD  #0          done yet?
20BC 26 EF          BNE   MOVE        branch if not

20BE 35 36          PULS  A,B,X,Y     restore regs
20C0 39             RTS               and return
```

```
                    *
                    * Name     - CLC
                    * Function - This routine compares two strings
                    *            as follows:
                    *
                    *
                    *              X -> string1
                    *              Y -> string2
                    *              A = length of strings to compare
                    *
                    *            String1 is compared to String2, and the
                    *            appropriate condition code is set.
                    *
                    *            Regs A,B,X,Y are preseved.
                    *
          20C1      CLC   EQU   *
20C1 34 36          PSHS  A,B,X,Y     save regs
          20C3      COMPR EQU   *
20C3 E6 80          LDB   0,X+        get a string1 char
20C5 E1 A0          CMPB  0,Y+        compare to string2 char
20C7 26 03          BNE   RPHOC       if not then done

20C9 4A             DECA              all through strings?
20CA 26 F7          BNE   COMPR       branch if not
```

```
                        200C  RPMOC   EQU   *
2CC 35  36                    PULS  A,B,X,Y   restore regs
2CE 39                        RTS           and return


                              *
                              * Name     - PDATA
                              * Function - This routine prints a string to an
                              *            output device. On entry, X points
                              *            to the string to print. As with the
                              *            PSTRNG routine in FLEX, an EOT
                              *            delimiter must appear after the
                              *            string to print.
                              *            Regs A and B will remain as is, and
                              *            X will be pointing to the EOT
                              *            delimiter char.
                              *
                        200F  PDATA   EQU   *
200F 34  06                   PSHS  A,B     save regs

                        2001  PDATA1  EQU   *
2001 A6  80                   LDA   0,X+    get char from string
2003 81  04                   CMPA  #EOT    end of text?
2005 27  05                   BEQ   PDATA2  branch if yes
2007 BD  CD1B                 JSR   PUTCHR  print char
200A 20  F5                   BRA   PDATA1  and continue

                        200C  PDATA2  EQU   *
200C 35  06                   PULS  A,B     restore regs
200E 39                       RTS           and return


                              *
                              * Name     - ROUTE
                              * Function - This routine determines where the user
                              *            wants his output to be routed. The user
                              *            has three options:
                              *
                              *            RO=TERMINAL: In this case, no control
                              *               info need be set, other than ensuring
                              *               the output switch (OSWTCH) is non
                              *               zero so FLEX will use the OUTCH2
                              *               routine to print characters.
                              *
                              *            RO=PRINTER: In this case, the printer
                              *               module routine must be loaded into
                              *               memory if it is not already there,
                              *               and the output switch must be cleared
                              *               so FLEX will use OUTCH to process
                              *               the characters
                              *
                              *            RO=FILE: In this case, the user is
                              *               prompted for a file to which all
                              *               output will be routed. FLEX's PUTCHR
                              *               routine automatically routes outputted
                              *               chars to the file's FCB if told
                              *               to do so.
                              *
                              *         All registers are preserved
                              *
                        200F  ROUTE   EQU   *
200F 34  36                   PSHS  A,B,X,Y save regs

20E1 8E  0000                 LDX   #0      clear file output address
20E4 BF  CC24                 STX   FOA     in FLEX

20E7 8E  0858                 LDX   #OUTFIL prompt user for route
20EA BD  CD1E                 JSR   PSTRNG
20ED BD  CD09                 JSR   INCH    get response
20F0 84  5F                   ANDA  #$5F    lowercase->uppercase

20F2 81  50                   CMPA  #'P     printer?
20F4 27  6F                   BEQ   R2      branch if so

20F6 81  54                   CMPA  #'T     terminal?
20F8 27  72                   BEQ   R3      branch if so

20FA 81  0D                   CMPA  #CR     CR? (same as terminal)
20FC 27  6E                   BEQ   R3      if so branch

20FE 81  46                   CMPA  #'F     file?
2100 26  71                   BNE   R6      leave if not

                              *
                              * output to a file
                              *
2102 8E  088F                 LDX   #TOFILE ask for "to" filename
2105 BD  CD1E                 JSR   PSTRNG
2108 BD  CD1B                 JSR   INBUF   get response
210B 8E  0A38                 LDX   #FIL.FCB "to" FCB
210E BD  CCCD                 JSR   SETFIL  validate filename
2111 24  08                   BCC   R4      check if file exists

2113 8E  0889                 LDX   #INVSPC invalid filename entered
2116 BD  CD1E                 JSR   PSTRNG
2119 20  58                   BRA   R6      return

                        211B  R4    EQU   *
211B 86  08                   LDA   #OUT    set extension
211D BD  CD33                 JSR   SETEXT
                              *
                              * Open the file
                              *
2120 86  02                   LDA   #XOWRIT open for write
2122 A7  84                   STA   FCBFC,X save in FCB
2124 BD  D406                 JSR   FMSCAL  call FMS
2127 27  08                   BEQ   R7      continue if file not there
                              *
                              * File already exists -> say so and return
                              *
2129 8E  0805                 LDX   #FXIST
212C BD  CD1E                 JSR   PSTRNG
212F 20  42                   BRA   R6      return

                        2131  R7    EQU   *
2131 8E  08F0                 LDX   #FTYPE  ask for file type
2134 BD  CD1E                 JSR   PSTRNG
2137 BD  CD09                 JSR   INCH    get response
213A 84  5F                   ANDA  #$5F    lowercase->uppercase

213C 5F                       CLRB
213D B1  54                   CMPA  #'T     text?
213F 27  11                   BEQ   R8      branch if so

2141 B1  0D                   CMPA  #CR     CR? (same as text)
2143 27  0D                   BEQ   R8      branch if so

2145 B1  42                   CMPA  #'B     binary?
2147 26  2A                   BNE   R6      bad leave if not

2149 C6  A0                   LDB   #OPN+BINRY set open-bin bits
214B 86  FF                   LDA   #$FF    type=binary
214D B7  0A76                 STA   FIL.FCB+FCBSCF set compress. flag
2150 20  02                   BRA   R1

                        2152  R8    EQU   *
2152 C6  C0                   LDB   #OPN+TEX set open-text bits
                        2154  R1    EQU   *
2154 8E  0A38                 LDX   #FIL.FCB set all o/p to file
2157 BF  CC24                 STX   FOA
215A 7F  CC22                 CLR   OSWTCH  o/p to the file
215D F7  087C                 STB   OPNFLG  save flag bits
2160 7F  CC04                 CLR   WIDTH   zero TTY width value
2163 20  0A                   BRA   R9      return

                        2165  R2    EQU   *
                              *
                              * perform setup for output to printer
                              *
2165 7F  CC22                 CLR   OSWTCH  o/p to aux. device
2168 8D  0E                   BSR   PRTSET  fetch printer module
216A 20  03                   BRA   R9      and return
                        216C  R3    EQU   *
                              *
                              * perform setup for output to terminal
                              *
216C 7C  CC22                 INC   OSWTCH

                        216F  R9    EQU   *
216F 1C  FE                   CLC           set good RC
```

```
2171 20 02          BRA   R10       and leave

        2173 A6     EQU   *
2173 1A 01          SEC             set bad RC
        2175 R10    EQU   *
2175 35 36          PULS  A,B,X,Y   restore regs
2177 39             RTS             and return


                    *
                    * Name     - PRTSET
                    * Function - This routine is called to load the
                    *            printer module if necessary,
                    *            and init the PIA, so that output can
                    *            be routed to the printer.
                    *
                    *       No registers are preserved
                    *
        2178 PRTSET EQU   *
2178 7F 0C09        CLR   PAU       disable pause feature
217B B6 CDE4        LDA   POUT      get 1st byte of space
217E 81 39          CMPA  #'9       is it 'RTS'?
2180 26 29          BNE   P1S       if not the loaded

                    *
                    * Load printer routine
                    *
2182 8E 095B        LDX   #PSYS     move in print rtn name
2185 10BE C844      LDY   #SYSFCB+FCBNAM into system FCB
2189 CC 0008        LDD   #11
218C 17 FF1C        LBSR  MVC
218F 7F C843        CLR   SYSFCB+FCBDRN check drive 0

2192 9E C840        LDX   #SYSFCB   point to system FCB
2195 86 01          LDA   #1(READ   open for read
2197 A7 84          STA   FCBFC,X
2199 BD D406        JSR   FMSCAL    call FMS
219C 27 05          BEQ   P1        branch if open ok

219E BD CD3F        JSR   RPTERR    report error
21A1 20 11          BRA   P2        and return

        21A3 P1     EQU   *

21A3 86 FF          LDA   #SUFMSC   set for binary read
21A5 A7 98 3B       STA   FCBSCF,X

21A8 BD C030        JSR   LOAD      load module

        21AB P1S    EQU   *
21AB BD CCC0        JSR   PRINIT    go init port
21AE BE CDE4        LDX   #POUT     get o/p address
21B1 BF CB10        STX   OUTCH+1   stuff in FLEX
        2184 P2     EQU   *
2184 39             RTS             return


                    *
                    * Name     - RRSET
                    * Function - This routine is called to reset the
                    *            FLEX output switch and close any file
                    *            that might be open through the FILFCB
                    *            FCB.
                    *
        2185 RESET  EQU   *
2185 86 01          LDA   #1
2187 B7 CCC2        STA   OSWTCH    reset output switch
218A B7 CC09        STA   PAU       enable pause
218D 7D 087C        TST   OPNFLG    is a file open?
2190 27 15          BEQ   RSX       return if not

                    * Output was going to file, close it
                    *
21C2 B6 0870        LDA   TTYWID    restore TTY width
21C5 B7 CC04        STA   WIDTH

21C8 8E 0A3B        LDX   #FILFCB   set FCB
21CB 86 04          LDA   #4CLOSE   close code
21CD A7 84          STA   FCBFC,X
21CF BD D406        JSR   FMSCAL    call FMS
21D2 27 03          BEQ   RSX       if ok return

21D4 BD CD3F        JSR   RPTERR    report error
        21D7 RSX    EQU   *
21D7 7F 087C        CLR   OPNFLG    indicate no file open
21DA 39             RTS             and return


                    END   START

0 ERROR(S) DETECTED

SYMBOL TABLE:

ADDBI  CD36  ASREAD 0001  ASWRIT 0002  BAC    0008  BADFIL 09A4
BADIN  0A1B  BAK    0005  BAS    0003  BCNT   0000  BELL   0007
BIN    0000  BINRY  0020  BLK3   108F  BLK4   10B1  BLK5   10BD
BLK6   1DA7  BLKCNT 0881  BLKLST 198A  BLKOUT 1D49  BLOCK  203D
BS     CD00  BSE    CD07  BSIZ5  0800  BSIZ8  0400  BUFFER 088A
BUFPNT CC14  BUFSIZ 087F  CMDCNT 205B  CLASS  CD21  CLC    20C1
CLN    CC1A  CLOCK  F700  CMD    0002  CMDFLG CC2B  COC    CC29
COLDS  CD00  COLUMN 0002  COMPR  20C3  CONFIG 087E  COPYIT 1DB8
CPMBLK 0010  CPMBS  0010  CPMCF  000C  CPMDIR 0000  CPMEL  0003
CPMEXT 0009  CPMFEF 2000  CPMHL  0008  CPMLEN 0020  CPMNAM 0001
CPMNL  0008  CPMRS2 000D  CPMSC  000F  CR     0000  CRLF   080A
CTR    207B  CURC   0C1B  DAT    0007  DDRV   DE00  DEL    CC01
DEPTH  CCD3  DIR    0009  DIRBEG 0000  DIREND 00E5  DIRMSG 0777
DIRTS  0005  DIVIDE 2050  DIVISR 207C  DOCMND CD4B  DOS    CC00
DP1    1DFF  DP2    1E0E  DRV    087B  DRVPRT 07FD  DSIZ1  0002
DSIZ2  0000  DTYPE  0918  EJECT  CC08  ENDCOP 07CF  ENDIR  0796
ENTADR 0000  ENV    CC2D  EOBUF  0883  EOF    001A  EOL    CC02
EOT    0004  ERR    1CA0  ERR02  1C9A  ESC    CC0A  ESCFR  CC16
EX1    1E40  EX2    1E52  EX3    1E73  EXTNTS 1E16  EXTSIZ 0050
FACP   0010  FADP   0040  FARP   0020  FAMP   0080  FCBAS  0002
FCBASE 0409  FCBCDA 0025  FCBCP  001E  FCBCRN 0020  FCBCUR 0408
FCBDI  0022  FCBDN  0003  FCBEDA 0013  FCBESB 0001  FCBFA  090F
FCBFC  0000  FCBFCD 0019  FCBFDD 0032  FCBFS  0015  FCBFSR 0017
FCBLEN 0140  FCBLP  001C  FCBNAM 0004  FCBNUB 0024  FCBRI  0023
FCBRS1 0010  FCBRS2 0018  FCBSB  0040  FCBSCF 0038  FCBSCR 0035
FCBSDA 0011  FCBVER 0435  FCDDAY 001A  FDIRTH 0019  FCDYR  0018
FIA    CC26  FIEF   CC3F  FILFCB 0A3B  FIVE   1C0D  FLEX   CD00
FMS    D400  FMSCAL D406  FMSCLS D403  FMSERR CC20  FMSINT D400
FOA    CC24  FOUND  1F35  FSMRAM 0002  FSYSSB 0000  FTYPE  09F0
FIIST  0805  G01    1FF7  GET1   1E95  GET2   1EA7  GET3   1EB9
GET4   1EA3  GETBLK 1FE2  GETCHR CD15  GETDRV 1E87  GETFIL CD2D
GETHEX CD42  GETIT  206D  GETD   10E6  GT1    1ED6  GT2    1EDA
GT3    1EE3  GTBAD  1EE7  GTGOOD 1EE9  GTSPEC 1EAC  HEAD   0100
INBUF  CD1B  INCH   CD09  INCH2  CD0C  INDEC  CD48  INDRV  0843
INFILE 082C  INTRQ  0966  INVSPC 0889  IOFLG  CC21  ISWTCH CC23
JT     1C56  LAD    CC1B  LCL    0000  LF     000A  LINEBUF C080
LOAD   C030  LPC    2029  LS1    0004  LS2    0003  LS3    0008
LS4    0003  LS5    0002  LS6    0003  LSTRM  CC11  M0     1C5E
M1     1C72  M2     1C7C  M3     1C87  MAP    CC00  MAPADR 0887
MAPLEN 0889  MAPPER 2029  MAPT85 2097  MAPTBB 207D  MEMEND CC2B
MENU   1C37  MENU   1C34  MOVE   20AD  MVC    20AB  MX     1C8A
NOB    1BF3  NULL   CC05  NXTCH  CD27  OFF5   0003  OFF8   0002
OPN    0080  OPNFLG 087C  OSWTCH 0C22  OUT    0008  OUTADR CDA5
OUTCH  CD0F  OUTCH2 CD12  OUTDEC CD39  OUTDRV 0772  OUTEXT 0824
OUTFIL 085B  OUTHEX CD3C  OUTNAM 0818  OUTSIZ 0000  P1     21A3
P1S    21AB  P2     2184  PAU    CC09  PB1    1F94  PB10   1F72
PB11   1F76  PB12   1F7A  PB13   1F59  PB14   1F5D  PB2    1FB9
PB3    1FDF  PB4    1FD9  PB5    1FCB  PB6    1F7F  PB7    1F4E
PB8    1F90  PB9    1F63  PCRLF  CDC4  PDI    1CF1  PDS    1D04
PDA    1D1B  PD7    1D00  PDB    1D27  PDATA  20CF  PDATA1 2001

PDATA2 200C  PDIR   1CAA  PDRIT  1D2B  POUT   CDE4  PRCHK  CCD8
PREVC  CC19  PRINIT CCC0  PROM   1C40  PRT    000A  PRTBLK 1F40
PRTSET 217B  PSTRNG CD1E  PSYS   095B  PUTCHR CD1B  R1     2154
R10    2175  R2     2165  R3     216C  R4     2118  R6     2173
R7     2131  R8     2152  R9     216F  RDERR  0947  READSS 20A1
REWTER CD06  RESET  2185  RET    1F39  ROUTE  200F  RPROC  20CC
RPTERR CD3F  RSTRIO CD2A  RSX    21D7  SBDATA 0044  SBLINK 0040
SBRS1  0042  SCFMSC 009F  SCFSC  0000  SCR    000B  SECBLK 0008
SECSIZ 0B85  SET5   2069  SETDON 2068  SETEXT CD33  SETPAU 1C92
SFA    C980  SIRDRE 0023  SIRDAY 0024  SIRFSB 0010  SIRFSE 001F
SIRFSS 0021  SIRLEN 0028  SIRMTH 0023  SIRNTS 0026  SIRNAM 0010
SIRTS  0003  SIRVOL 0018  SIRYR  0025  SP     0020  SP3    0B2B
SP5    C700  SRC2   1F14  SRC3   1F27  SRCHM  1EEC  SSIZ5  0100
SSIZB  0080  START  1BDA  START1 1BDD  STAT   CD4E  STKA   C000
STSIZ  1C1B  SVEFIL 0000  SVDR   CC0E  SYDRV  CC0B  SYS    0004
SYSCON CD4E  SYSCR1 CC0D  SYSCR2 CC2A  SYSCR3 CC30  SYSCR4 CCF8
SYSFCB C840  TAB    CC06  TABCON 0009  TBLEN5 000A  TBLEN8 001A
TBLOFF 207A  TEx    0040  TMP1   0000  TMP3   0001  TOFILE 0B8F
```

```
TRADDR CC1E   TRFLG  CC1D   TTYWID 0078   TXT    0001   UCA    C100
UCTA   CC12   UNK1   1D3E   UNKOUT 1D35   URAM   0000   WARMS  CD03
WELCO2 0368   WELCOM 0126   WIDTH  CCD4   WKDRV  CCDC   XBOR   0016
XCLOSE 0004   XDELET 000C   XFWD   0014   XGIR   0007   XGRB   0011
XNSS   000F   XOD19  0006   XOREAD 0001   XOSIR  0010   XOUPUT 0003
XOWRIT 0002   XPIR   0008   XPURGE 0015   XPRS   0012   XREWAN 000C
XRES1  0008   XRES2  000E   XRES3  0013   XREWND 0005   XRSS   0009
XRWND  0000   XTMP   0001   XWSS   000A
```

```
*****************************************
*                                       *
*    DOS to FLEX Utility                *
*                                       *
*       Written by: Scott R. Fraser     *
*                   547 Sharron Bay     *
*                   Winnipeg, Manitoba, Canada  *
*                   R2G 0H8             *
*                   Ph. (204) 338-7641  *
*****************************************

*
* NOTE: Program tested for single sided,
*       single density disks only
*****************************************
*
* Uses SSB's DOS690 version
*

*
* DSECT for a DOS Sequential File
* Information Block (FIB)
*
0000                    ORG   $0000
0000 0018       FIB     EQU   *
0000                    FIBFN   RMB   6       file name
0006                    FIBEXT  RMB   3       file extension
0009                    FIBFT   RMB   1       file type
000A                    FIBPS   RMB   1       file status
000B                    FIBFSU  RMB   2       1st file sector used
000D                    FIBLSU  RMB   2       last file sector used
000F                    FIBSLC  RMB   2       file size (# sectors)
0011                    FIBRSV  RMB   6       reserved
0017                    FIBEFS  RMB   1       extended file status
0018            FIBLEN  EQU   *       length of FIB

*
* Define DOS directory DSECT
*
0000                    ORG   $0000
0000 0000       DOSDIR  EQU   *
0000                    DOSNXT  RMB   2       link of next dir block
0002                    DOSPRV  RMB   2       link of previous dir block
0004                    DOSDSS  RMB   1       dir sector size (preallocd)
0005                    DOSUNU  RMB   3       unused ($00)
0008            DOSLEN  EQU   *       length of dsect

*
* Define DOS DSECT for free space list info
*  (starts at offset +8, trk 00 sec 01)
*
0000                    ORG   $0000

0000 0000       DOSFSL  EQU   *
0000                    FSLFF   RMB   1       must be $FF
0001                    FSLDC   RMB   8       don't cares ($FF)
0009                    FSLDIF  RMB   1       disk interleaving factor
000A                    FSLDT   RMB   1       disk type
000B                    FSLNXT  RMB   2       link of next free block
000D                    FSLLST  RMB   2       trk/sec of last free block
000F                    FSLSC   RMB   2       count of available sectors
0011                    FSLBOT  RMB   2       trk/sec of boot file
0013                    FSLSPD  RMB   2       count of # sectors/disk
0015                    FSLUNU  RMB   3       unused
0018            FSLLEN  EQU   *       length of FSL

0100                    ORG   $0100
*
***** DEFINE PROMPTS USED BY ALL ROUTINES
*
*
* PDIR routine
*
0100 0100       HEAD    EQU   *
0100 000A               FDB   CRLF
0102 4E 41 4D 45        FCC   /NAME  TYPE  BEGIN   END/
0119 20 20 20 20
0121 000A               FDB   CRLF
0123 04                 FCB   EOT

0124 0124       WELCOM  EQU   *
0124 000A               FDB   CRLF
0126 44 65 73 63        FCC   /Description of Selections:/
0140 000A 000A          FDB   CRLF,CRLF
0144 20 20 20 30        FCC   /   0 - you're looking at it buddy!/
0166 000A               FDB   CRLF
0168 20 20 20 31        FCC   /   1 - this selection will yield a DOS /
018F 64 69 72 65        FCC   /directory listing./
01A1 000A               FDB   CRLF
01A3 20 20 20 20        FCC   /       The user is prompted for the drive/
01CC 20 6F 66 20        FCC   / of the DOS disk/
01DC 000A               FDB   CRLF
01DE 20 20 20 32        FCC   /   2 - this  selection  will  copy  a  /
0205 67 69 76 65        FCC   /given DOS file to a/
0218 000A               FDB   CRLF
021A 20 20 20 20        FCC   /       FLEX  file.   The  user is/
0238 20 70 72 6F        FCC   / prompted for the "from" drive/
0259 000A               FDB   CRLF
025B 20 20 20 20        FCC   /       (ie. where the DOS disk is)/
0279 20 61 6E 64        FCC   / and the "from" file/
0291 000A               FDB   CRLF
0293 20 20 20 20        FCC   /       name.  No conversion (ear/
02B3 20 6C 6F 77        FCC   / lowercase -> uppercase/
02CA 000A               FDB   CRLF
02CC 20 20 20 20        FCC   /       for example) is done on this/
02EF 20 6E 61 6D        FCC   / name, so type it/

0300 000A               FDB   CRLF
0302 20 20 20 20        FCC   /       exactly as it appears in your/
0326 20 64 69 72        FCC   / directory listing./
0339 000A               FDB   CRLF
033B 20 20 20 33        FCC   /   3 - exit program/
034E 000A 000A          FDB   CRLF,CRLF
0352 4D 6F 72 65        FCC   /More to Come ...../
0369 04                 FCB   EOT
036A 036A       WELCO2  EQU   *
036A 000A 000A          FDB   CRLF,CRLF
036E 4E 6F 74 65        FCC   /Note:  a) the user may hit [Return]/
0391 20 69 6E 73        FCC   / instead of typing a/
03A5 000A               FDB   CRLF
03A7 20 20 20 20        FCC   /         drive number.  In this case/
03CC 20 74 68 65        FCC   / the last specified/
03DF 000A               FDB   CRLF
03E1 20 20 20 20        FCC   /         drive number will be used/
0404 20 28 6F 72        FCC   / (or the FLEX work/
0416 000A               FDB   CRLF
0418 20 20 20 20        FCC   /         drive in the case no previous/
043F 20 64 72 69        FCC   / drive was given)/
0450 000A 000A          FDB   CRLF,CRLF
0454 20 20 20 20        FCC   /       b) the user has the option of/
0478 20 72 6F 75        FCC   / routing his output/
0488 000A               FDB   CRLF
048A 20 20 20 20        FCC   /         from selection 1 or 2, to/
04B0 20 74 68 65        FCC   / the printer, terminal./
04C7 000A               FDB   CRLF
04C9 20 20 20 20        FCC   /         or a file.  If "printer"/
04EB 20 69 73 20        FCC   / is selected and the/
04FF 000A               FDB   CRLF
0501 20 20 20 20        FCC   /         FLEX printer module hasn't/
0523 20 62 65 65        FCC   / been loaded, it/
0535 000A               FDB   CRLF
0537 20 20 20 20        FCC   /         (PRINT.SYS) will/
0551 20 61 75 74        FCC   / automatically be loaded/
0569 000A               FDB   CRLF
056B 20 20 20 20        FCC   /         from drive 0.  If "file"/
0589 20 69 73 20        FCC   / is selected, the user/
05A3 000A               FDB   CRLF
05A5 20 20 20 20        FCC   /         will be prompted for the/
05C7 20 22 74 6F        FCC   / "to" file name.  Type/
05DD 000A               FDB   CRLF
05DF 20 20 20 20        FCC   /         this as a standard FLEX/
```

```
0600 20 66 69 6C          FCC   / file specification (es:/
0618 0D0A                 FDB   CRLF
061A 20 20 20 20          FCC   /         0.JUNK.TXT).  The usee/
063A 20 77 69 6C          FCC   / will also be asked/
0640 0D0A                 FDB   CRLF
064F 20 20 20 20          FCC   /         whether the file is/
066C 20 22 74 65          FCC   / "text" or "binary". If/
0683 0D0A                 FDB   CRLF
0685 20 20 20 20          FCC   /         type "binary" is chosen,/
06A7 20 74 68 65          FCC   / the FLEX space/
0686 0D0A                 FDB   CRLF
0688 20 20 20 20          FCC   /         compression flag will be/
06DA 20 73 65 74          FCC   / set and no expansion/
06EF 0D0A                 FDB   CRLF
06F1 20 20 20 20          FCC   /         of tabs, etc. is performed./
0716 20 20 49 66          FCC   /  If ROUTE=P or T./
0728 0D0A                 FDB   CRLF
072A 20 20 20 20          FCC   /         tabs will be expanded./
074A 0D0A 0D0A            FDB   CRLF,CRLF
074E 45 6E 74 65          FCC   /Enter (Return) to Continue.../
076B 04                   FCB   EOT

            076C  DIRMSG  EQU   *
076C 0D0A                 FDB   CRLF
076E 20 20 20 20          FCC   /     Directory of Drive /
0787 00             OUTDRV FCB  0       put drive# here
0788 0D0A                 FDB   CRLF
078A 04                   FCB   EOT

            078B  ENDIR   EQU   *
078B 20 20 20 45          FCC   /  End of Directory Listing. /
07A8 48 69 74 20          FCC   /Hit (Return) to Continue.../
07C3 04                   FCB   EOT

            07C4  ENDCOP  EQU   *
07C4 20 20 20 45          FCC   /  End of Copy. Enter /
07DA 5B 52 65 74          FCC   /(Return) to Continue.../
07F1 04                   FCB   EOT

            07F2  DRVPMT  EQU   *
07F2 20 20 20 45          FCC   *  Enter drive of DOS disk: *
080E 04                   FCB   EOT

                  *
                  * BLNOUT routine
                  *
080F 20 20 20 20    SP4   FCC   /   /    4 spaces
0813 04                   FCB   EOT

                  *
                  * COPYIT routine
                  *
            0814  INFILE  EQU   *
0814 20 20 20 45          FCC   /  Enter "from" file: /
082A 04                   FCB   EOT

            0828  INDRV   EQU   *
0828 20 20 20 45          FCC   /  Enter "from" drive: /
0842 04                   FCB   EOT

            0843  OUTFIL  EQU   *
0843 20 20 20 4F          FCC   /  Output to File(F), Printer(P), /
0865 6F 72 20 54          FCC   /or Terminal(Te)? /
0876 04                   FCB   EOT

            0877  TOFILE  EQU   *
0877 20 20 20 45          FCC   /  Enter "to" file: /
0888 04                   FCB   EOT
            088C  BADFIL  EQU   *
088C 46 49 4C 45          FCC   /FILE DOES NOT EXIST!/
08A0 04                   FCB   EOT

            08A1  INVSPC  EQU   *
08A1 49 4E 56 41          FCC   /INVALID FILE SPECIFICATION!/
08BC 04                   FCB   EOT

            08BD  FXIST   EQU   *
08BD 53 4F 52 52          FCC   /SORRY, FILE ALREADY EXISTS/
08D7 04                   FCB   EOT

            08D8  FTYPE   EQU   *
08D8 20 20 20 49          FCC   /  Is file type Binary(B) /
08F2 6F 72 20 54          FCC   /or Text(Te)? /
08FF 04                   FCB   EOT

                  *
                  * Misc
                  *
0900 3C 3C 3C 20    RDERR FCC   /<<< READ ERROR >>>!/
0913 04                   FCB   EOT

0914 50 53 49 4E    PSYS  FCC   /PRINT/
0919 00 00 00             FCB   0,0,0
091C 53 59 53             FCC   /SYS/

            091F  INTRO   EQU   *
091F 0D0A 0D0A            FDB   CRLF,CRLF
0923 20 20 20 46          FCC   *   FLEX/DOS UTILITY (C) 1983*
093F 20 62 79 20          FCC   * by Scott Fraser*
094F 0D0A 0D0A            FDB   CRLF,CRLF
0953 20 20 20 20          FCC   /         0 - Directions/
0968 20 20 20 20          FCC   *         1 - DOS Directory*
0986 0D0A                 FDB   CRLF
0988 20 20 20 20          FCC   /         2 - Copy File/
099F 20 20 20 20          FCC   /          3 - EXIT Program/
09BA 0D0A 0D0A            FDB   CRLF,CRLF
09BE 20 20 20 45          FCC   /  ENTER SELECTION: /
09D2 04                   FCB   EOT

            09D3  BADIN   EQU   *
09D3 20 20 20 49          FCC   /  INVALID SELECTION. REENTER: /
09F2 04                   FCB   EOT

09F3              FILFCB  RMB   FCBLEN    reserve FCB area

                  *
                  * Define some DOS constants
                  *
            0080  SSIZ5   EQU   128       #bytes per sector (5")
            0000  EOF     EQU   $00       delimits end of file
            00FE  DIREND  EQU   $FE       indicates end of directory
            0001  DIRSEC  EQU   $0001     trk/sec of start of DOS dir
            0009  TABCHR  EQU   $09       universal tab char ((cntl)I)

0833              DRV     RMB   1         holds current drive#
0834 00           OPNFLG  FCB   0         file open bits
            0080  OPN     EQU   %10000000 #file is open
            0040  TEX     EQU   %01000000 #file is text
            0020  BINRY   EQU   %00100000 #file is binary
0835              TTYWID  RMB   1         save TTY Width value here

            0080  BSIZ5   EQU   SSIZ5*1   size of buffer (5")

0836              BUFFER  RMB   BSIZ5     alloc max buffer size



C100                      ORG   $CA
C100 20   01        START BRA   START1    ***** Begin Program *****

C102 04                   FCB   4         <<<< VERSION # >>>>



            C103  START1  EQU   *
C103 10FE CE2B            LDS   #STEND    initialize system stack ptr

C107 B6   CC0C            LDA   #WKDRV    default to work drive
C10A B7   0833            STA   DRV       save default
C10D 8E   C119            LDX   #MENU
C110 BF   CC16            STX   ESCRR     modify escape return register
C113 B6   CC04            LDA   WIDTH     get current TTY line width
C116 B7   0835            STA   TTYWID    and save it

            C119  MENU    EQU   *
C119 7F   CC1A            CLR   CLN       clear current line num ist
            C11C  MENU2   EQU   *
C11C 7C   CC22            INC   OSWTCH    set output switch
C11F 8E   091F            LDX   #INTRO    print intro msg
```

```
C122 BD  CD1E          JSR   PSTRNG   ask for selection

         C125  PROM    EQU   *
C125 BD  CD09          JSR   INCH     get a response
C128 BD  CD21          JSR   CLASS    classify it
C12B 25  42            BCS   MI       bad selection
C12D 81  33            CMPA  #'3      over 3?
C12F 22  3E            BHI   MI       bad selection

C131 84  0F            ANDA  #%00001111 keep low 4 bits only
C133 48                ASLA           selection=selection*2
C134 8E  C138          LDX   #JT      addr of jump table
C137 30  86            LEAX  A,X      addr proper selection
C139 6E  84            JMP   0,X      goto selection

C13B 20  06      JT    BRA   M0       selection=0
C13D 20  18            BRA   M1       =1
C13F 20  20            BRA   M2       =2
C141 20  29            BRA   M3       =3

         C143  M0      EQU   *        do selection 0
C143 8E  0124          LDX   #MEL.COM print instructions
C146 BD  CD1E          JSR   PSTRNG

C149 B6  CC03          LDA   DEPTH    breather
C14C B7  CC1A          STA   CLN
C14F 8E  036A          LDX   #MEL.CO2 more to come
C152 BD  CD1E          JSR   PSTRNG
C155 20  20            BRA   SETPAU

         C157  M1      EQU   *
C157 BD  32            BSR   PDIR     print dir
C159 8E  0788          LDX   #ENDDIR
C15C BD  CD1E          JSR   PSTRNG
C15F 20  16            BRA   SETPAU

         C161  M2      EQU   *
C161 17  00ED          LBSR  COPY1T   copy file
C164 8E  07C4          LDX   #ENDCOPY
C167 BD  CD1E          JSR   PSTRNG
C16A 20  0B            BRA   SETPAU

         C16C  M3      EQU   *
C16C 7E  CD03          JMP   WARMS    return to FLEX

         C16F  MX      EQU   *        bad selection
C16F 8E  0903          LDX   #BADIN
C172 BD  CD1E          JSR   PSTRNG   print mss
C175 20  AE            BRA   PROM

         C177  SETPAU  EQU   *
C177 B6  CC03          LDA   DEPTH    pause (if set)
C17A B7  CC1A          STA   CLN      so user can read stuff
C17D 20  9D            BRA   MEN2

         C17F  ERR02   EQU   *
C17F BD  CD3F          JSR   RPTERR   report error first
C182 8E  0900          LDX   #ORDERR  set mss

         C185  ERR     EQU   *
C185 BD  CD1E          JSR   PSTRNG   print mss
C188 7E  CD03          JMP   WARMS    and return to FLEX


                       *
                       * Name      - PDIR
                       * Function  - This routine is called to print out
                       *            directory information on a DOS disk.
                       *
                       *            No parms are required
                       *            All registers are used and not restored
                       *

         C18B  PDIR    EQU   *
C18B 32  E9 FFFC       LEAS  -LS1,S   alloc local storage

C18F 8E  07F2          LDX   #DRVPMT  ask for a drive
C192 BD  CD1E          JSR   PSTRNG
C195 17  0112          LBSR  GETDRV   set drive #
C198 1025 0093         LBCS  PDXIT    if bad then leave

C19C 17  0284          LBSR  ROUTE    where to route output
C19F 1025 008C         LBCS  PDXIT    leave if bad
C1A3 BD  CD24          JSR   PCRLF
C1A6 B6  0833          LDA   DRV      set drive
C1A9 B7  C843          STA   SYSFCB+FCBDN save in FCB
C1AC 8A  30            ORA   #'0      convert to ascii
C1AE B7  0787          STA   OUTDRV   save to print

C1B1 8E  076C          LDX   #DIRMSG  print mss
C1B4 17  025C          LBSR  PDATA

C1B7 8E  0100          LDX   #HEAD    print dir header
C1BA 17  0256          LBSR  PDATA

C1BD CC  0001          LDD   #DIRBEG  set beginning of directory

         C1C0  PD3     BSR   *
C1C0 17  0202          LBSR  GETBLK   get a block
C1C3 ED  E9 0002       STD   TEMP,S   save next trk/sec to read
                       *
                       * BUFFER is full of directory data. If
                       * 1st sector of directory then skip over
                       * dir data and FSL stuff, otherwise just
                       * skip over dir data.
                       *
C1C7 8E  0836          LDX   #BUFFER  position to 1st entry
C1CA EC  02            LDD   DCSPRV,X 1st sector of dir?
C1CC 26  03            BNE   PD2      branch if not
C1CE 30  88 18         LEAX  FSLLEN,X skip over FSL info
         C1D1  PD2     BSR   *
C1D1 30  08            LEAX  DOSLEN,X skip over dir info

         C1D3  PD1     EQU   *
C1D3 AF  E9 0000       STX   ENTADR,S save entry addr

C1D7 A6  84            LDA   FIBFN,X  valid file entry?
C1D9 27  42            BEQ   PD4      branch if not
C1DB 81  FE            CMPA  #DIRDEND end of dir yet?
C1DD 27  50            BEQ   PDXIT    branch if so

                       *
                       * Now print the file name
                       *
C1DF 30  84            LEAX  FIBFN,X  field to move
C1E1 C6  09            LDB   #9       length to move
         C1E3  PD5     EQU   *
C1E3 A6  80            LDA   0,X+     get a char
C1E5 26  02            BNE   *+2+2    if =0 then skip
C1E7 86  20            LDA   #SP
C1E9 BD  CD18          JSR   PUTCHR   print the char
C1EC 5A                DECB
C1ED 27  08            BEQ   PD6      exit if all done
C1EF C1  03            CMPB  #3       just extension left?
C1F1 26  F0            BNE   PD5      branch if not
C1F3 B6  2E            LDA   #'.      print a '.' first
C1F5 BD  CD18          JSR   PUTCHR
C1F8 20  E9            BRA   PD5      and continue

         C1FA  PD6     BSR   *
C1FA AE  E9 0000       LDX   ENTADR,S
C1FE 30  08            LEAX  FIBFSU,X print 1st file sector
C200 BD  35            BSR   LNKOUT

C202 AE  E9 0000       LDX   ENTADR,S
C206 30  00            LEAX  FIBLSU,X print last file sector
C208 BD  20            BSR   LNKOUT

C20A 8E  000F          LDX   #SP4
C20D 17  0203          LBSR  PDATA    print 4 spaces
C210 AE  E9 0000       LDX   ENTADR,S
C214 30  0F            LEAX  FIBSUC,X print file size
C216 5F                CLRB
C217 BD  CD39          JSR   OUTDEC
C21A BD  CD24          JSR   PCRLF

         C21D  PD4     EQU   *
C21D AE  E9 0000       LDX   ENTADR,S
C221 30  88 18         LEAX  FIBLEN,X goto next file entry
C224 8C  0B0A          CMPX  #BUFFER+BSIZE finished?
C227 25  AA            BLO   PD1      branch if not
```

```
C229 EC  E9 0002        LDD   TEMP,S    read in next sector
C22D 26  91             BNE   P03       and continue

         C22F PDXIT     EQU   *
C22F 17  02C7           LBSR  RESET     clear things up first

C232 32  E9 0004        LEAS  LS1,S     release local storage
C236 39                 RTS             return

                *
                * Data area for PDIR routine
                *
         C237   LPC     SET   *
0000                    ORG   $0000
         0000   LCL     SET   *
0000     ENTADR RMB   2
0002     TEMP   RMB   2
         0004   LS1     EQU   *-LCL     len of local area

C237                    ORG   LPC       restore PC


                *
                * Name     - LNKOUT
                * Function - This routine takes in a pointer to
                *            a 2 byte hex link field (ptr in X)
                *            and prints it as 2 one byte fields
                *            separated by a "-".
                *
                *            All registers are preserved
                *
         C237   LNKOUT  EQU   *
C237 34  12             PSHS  A,X       save registers
C239 86  20             LDA   #SP       print a space first
C23B BD  CD18           JSR   PUTCHR
C23E BD  CD18           JSR   PUTCHR

C241 BD  CD3C           JSR   OUTHEX
C244 30  01             LEAX  1,X
C246 86  2D             LDA   #'-
C248 BD  CD18           JSR   PUTCHR
C24B BD  CD3C           JSR   OUTHEX

C24E 35  12             PULS  A,X       restore registers
C250 39                 RTS             and return


                *
                * Name     - COPVIT
                * Function - This rtn is called to copy a DOS file
                *            to a FLEX file. The user is prompted
                *            for a "from" file (off the DOS disk)
                *            and a "to" file (in FLEX). If the "to"
                *            file is not specified, output goes to
                *            the terminal.
                *
                *            All registers are used and not restored
                *
         C251   COPVIT  EQU   *
C251 32  E9 FFF7        LEAS  -LS3,S    allec local storage

C255 8E  0820           LDX   #INDRV    set "from" drive
C258 BD  CD1E           JSR   PSTRNG
C25B BD  4D             BSR   GETDRV    set drive#
C25D 25  43             BCS   DP2       if bad then leave

C25F 8E  0814           LDX   #INFILE   prompt for "from" filename
C262 BD  CD1E           JSR   PSTRNG
C265 BD  CD1B           JSR   INBUF     get "from" filename
C268 30  E9 0000        LEAX  SVEFIL,S  where to put filename
C26C BD  61             BSR   GTSPEC    get DOS filename
C26E 25  32             BCS   DP2       if bad then leave

C270 17  009C           LBSR  SRCHR     go do search
C273 24  08             BCC   GET0      if found, get "to" file
                *
                * File not found in DOS directory -> flag
                * and reprompt user.
                *
C275 8E  089C           LDX   #BADFIL   print mss
C278 BD  CD1E           JSR   PSTRNG

C27B 20  25             BRA   DP2       return

                *
         C27D   GET0    EQU   *
                *
                * The DOS directory entry has been found.
                * Get the beginning track/sector of the
                * file and continue transfer until end link
                * found
                *

C27D B6  0833           LDA   DRV       get drive #
C280 B7  C843           STA   SYSFCB+FCBDN save in FCB
C283 17  019D           LBSR  ROUTE     get route
C286 7D  0834           TST   OPNFLG    writing to a file?
C289 26  03             BNE   DP1       yes, then no lead CRLF

C28B BD  CD24           JSR   PCRLF     print CRLF first
         C28E   DP1     EQU   *
C28E EC  08             LDD   F1&FSU,X  get start of file
C290 84  3F             ANDA  #%00111111 keep low 6 bits of trk
C292 C4  1F             ANDB  #%00011111 keep low 5 bits of sec
         C294   DP4     EQU   *
C294 1083 0000          CMPD  #$0000    end of file?
C298 27  08             BEQ   DP2       branch if so

C29A 17  012B           LBSR  GETBLK    get a block of data
C29D 17  00BF           LBSR  PRTBLK    and print it
C2A0 20  F2             BRA   DP4       continue

         C2A2   DP2     EQU   *
C2A2 17  0254           LBSR  RESET     clear things up first

C2A5 32  E9 0009        LEAS  LS3,S     release local storage
C2A9 39                 RTS             return

                *
                * Data area for COPVIT routine
                *
         08AA   LPC     SET   *
0000                    ORG   $0000
         0000   LCL     SET   *
0000     SVEFIL RMB   9
         0009   LS3     EQU   *-LCL     len of local storage

C2AA                    ORG   LPC       restore PC


                *
                * Name     - GETDRV
                * Function - This routine gets a drive # from the
                *            terminal. The user can either enter a
                *            drive # in the range 0-3, or type null,
                *            in which case the default drive# in
                *            "DRV" will be taken.
                *
                *            The carry is clear if # ok, else it is
                *            set. The valid drive number is returned
                *            in ACC A, and is also saved in DRV.
                *
                *            All registers are preserved.
                *
         C2AA   GETDRV  EQU   *
C2AA 34  34             PSHS  B,X,Y     save regs

C2AC BD  CD09           JSR   INCH      get response
C2AF 81  00             CMPA  #CR       use default?
C2B1 26  05             BNE   GET1      branch if not

C2B3 B6  0833           LDA   DRV       get default
C2B6 20  0E             BRA   GET4      and save it

         C2B8   GET1    EQU   *
C2B8 BD  CD21           JSR   CLASS     classify it
C2BB 25  00             BCS   GET2      bad drive
C2BD 81  33             CMPA  #'3       in range 0-3?
C2BF 22  09             BHI   GET2      bad drive

C2C1 84  0F             ANDA  #%00001111 keep low 4 bits
C2C3 B7  0833           STA   DRV       save in DRV storage
         C2C6   GET4    EQU   *
```

```
C2C6 1C FE              CLC             set good return
C2C8 20 02              BRA     GET3    and return

        C2CA GET2       EQU     *
C2CA 1A 01              SEC             set bad return
        C2CC GET3       EQU     *
C2CC 36 34              PULS    B,X,Y   restore regs
C2CE 39                 RTS             and return


                *
                * Name      - GTSPEC
                * Function - This routine parses the system INBUF
                *           buffer for a DOS file name. A valid
                *           filename must contain up to 6 chars,
                *           then an optional '.' and up to 3 char
                *           file extension. The X reg points to
                *           an area to place the 9 char file name
                *           and any unused chars are padded with
                *           spaces.
                *
                *           Carry is clear if file spec ok, else
                *           it is set.
                *
                *           All regs are preserved.
                *
        C2CF GTSPEC     EQU     *
C2CF 34 36              PSHS    A,B,X,Y save regs

                *
                * First blank out the "to" field
                *
C2D1 86 00              LDA     #0      pad with zeroes
C2D3 A7 84              STA     0,X
C2D6 31 01              LEAY    1,X     "to" field
C2D7 CC 0008            LDD     #9-1    length to move
C2DA 17 0112            LBSR    MVC     move in spaces

C2DD 108E C080          LDY     #LINEBUF pt to system buffer
C2E1 86 09              LDA     #9      # chars to move

C2E3 E6 A4              LDB     0,Y     user just hit return?
C2E5 C1 00              CMPB    #CR
C2E7 27 21              BEQ     GTBAD   yes, then bad return
        C2E9 GT1        EQU     *
C2E9 E6 A0              LDB     0,Y+    get a char

C2EB C1 20              CMPB    #SP     space here?
C2ED 27 FA              BEQ     GT1     skip spaces

C2EF C1 2E              CMPB    #'.     have an extension
C2F1 26 0A              BNE     GT2     nope, then continue
                *
                * The rest of the filename from the system
                * buffer is the extension. Adjust the
                * "to" ptr so that any unused chars in the
                * "to" name are spaces
                *
C2F3 1F 89              TFR     A,B     copy A
C2F5 C0 03              SUBB    #3
C2F7 2B 11              BMI     GTBAD   if neg then bad name
C2F9 30 85              LEAX    B,X     adjust "to" ptr
C2FB 20 EC              BRA     GT1     and continue

        C2FD GT2        EQU     *
C2FD C1 00              CMPB    #CR     end yet?
C2FF 27 05              BEQ     GT3     done if so

C301 E7 80              STB     0,X+    save to "to" field
0003 4A                 DECA            moved all chars?
C304 26 E3              BNE     GT1     branch if not

        C306 GT3        EQU     *
C306 1C FE              CLC             good return
C308 20 02              BRA     GTGOOD
        C30A GTBAD      EQU     *
C30A 1A 01              SEC             set bad return
C30C 35 36      GTGOOD  PULS    A,B,X,Y restore regs
C30E 39                 RTS             and return


                *
                * Name      - SRCHM
                * Function - This routine searches the directory on
                *           a DOS disk for the file name pointed to
                *           by the X register. "SYSFCB" is the FCB
                *           used, and is assumed to contain the
                *           drive where the DOS disk lies
                *
                *           On exist, X->DOS file entry of found
                *           file. If file is not found, the carry
                *           is set, otherwise it is clear
                *
                *           All registers are preserved
                *
        C30F SRCHM      EQU     *
C30F 34 26              PSHS    A,B,Y   save regs
C311 32 E9 FFFE         LEAS    -LSS,S  alloc local storage

C315 CC 0001            LDD     #DIRBEG start with 1st dir sector

        C318 SRC1       EQU     *
C318 17 00AA            LBSR    GETBLK  get a block
C31B ED E9 0000         STD     TMP4,S  save next sector to read
C31F 108E 0B36          LDY     #BUFFER Y-> dir data
C323 EC 22              LDD     DOSPRV,Y is this 1st dir sector?
C325 26 03              BNE     SRC3    branch if not
C327 31 A8 18           LEAY    FSLLEN,Y skip over FSL info
        C32A SRC3       EQU     *
C32A 31 28              LEAY    DOSLEN,Y skip over dir info
        C32C SRC2       EQU     *
C32C A6 A4              LDA     FIBFN,Y get 1st char of file name
C32E 27 11              BEQ     SRC4    goto next entry if so
C330 81 FE              CMPA    #DIREND end of directory?
C332 27 1C              BEQ     SRCXIT  branch if so

                *
                * Compare dir file with file set down to
                * see if a match
                *
C334 34 20              PSHS    Y       save entry ptr

C336 31 A4              LEAY    FIBFN,Y Y->string2
C338 86 09              LDA     #9      len to compare
C33A 17 00C8            LBSR    CLC     compare

C33D 35 20              PULS    Y       restore entry ptr
C33F 27 13              BEQ     FOUND   branch if a match

        C341 SRC4       EQU     *
C341 31 A8 18           LEAY    FIBLEN,Y go to next entry
0044 108C 0B86          CMPY    #BUFFER+BSIZS finished?
C348 25 E2              BLO     SRC2    branch if not

C34A EC E9 0000         LDD     TMP4,S  goto next dir sector
C34E 26 C8              BNE     SRC1    and continue search

        C350 SRCXIT     EQU     *
C350 1A 01              SEC             file not found
C352 20 04              BRA     RET     return

        C354 FOUND      EQU     *
C354 1C FE              CLC             file found
C356 1F 21              TFR     Y,X     X-> found entry
        C358 RET        EQU     *

C358 32 E9 0002         LEAS    LSS,S   release local storage
C35C 35 26              PULS    A,B,Y   restore regs
C35E 39                 RTS             and return
                *
                * Data area for SRCHM routine
                *
        C35F LPC        SET     *
0000                    ORG     $0000
        0000 LCL        SET     *
0000            TMP4    RMB     2
        0002 LSS        EQU     *-LCL   len of local storage

C35F                    ORB     LPC     restore PC
```

```
                    * Name     - PRTBLK
                    * Function - This routine dumps the contents of
                    *            BUFFER to the standard output.
                    *
                    *            All registers are preserved
                    *
                    *
          C35F      PRTBLK     EQU   *
C35F 34   36                   PSHS  A,B,I,Y     save regs

C361 8E   0B36                 LDX   #BUFFER     point to buffer area
C364 30   04                   LEAX  4,X         skip over link info
C366 5F                        CLRB              start column counter at 0
          C367      PB1        EQU   *
C367 8C   ????                 CMPX  #BUFFER+BSIZE end of buffer yet?
C36A 27   56                   BEQ   PB3         branch if so
C36C A6   80                   LDA   0,X+        get a char

C36E 34   04                   PSHS  B           save col ctr
C370 F6   0B34                 LDB   OPNFLG      get open flag bits
C373 C1   A0                   CMPB  #OPN+BINRY  file open and binary?
C375 35   04                   PULS  B           restore col ctr
C377 27   43                   BEQ   PB4         branch if so

C379 81   0A                   CMPA  #LF         line feed?
C37B 27   EA                   BEQ   PB1         skip if so

C37D 81   0D                   CMPA  #CR         a carriage return?
C37F 26   08                   BNE   PB2         branch if not

C381 5F                        CLRB              clear column counter
C382 7D   0B34                 TST   OPNFLG      writing to a file?
C385 26   35                   BNE   PB4         branch if so
C387 BD   CD24                 JSR   PCRLF       print a CRLF
C38A 20   08                   BRA   PB1
          C38C      PB2        EQU   *
C38C 81   09                   CMPA  #TABCON     a tab char?
C38E 26   1C                   BNE   PB6         branch if not

                    *
                    * Encountered a tab character. If outputting
                    * to a file, the tab must be converted to the
                    * proper spaces, depending on the current
                    * column.
                    *
C390 34   04                   PSHS  B           save current column ctr
C392 C4   07                   ANDB  #%00000111  B = mod(curcol,8)
C394 86   08                   LDA   #8
C396 34   04                   PSHS  B
C398 A0   E0                   SUBA  0,S+        A = 8 - mod(curcol,8)
C39A 34   02                   PSHS  A           save this value too
C39C 1F   89                   TFR   A,B         B = #spaces to insert

C39E 86   20                   LDA   #SP
          C3A0      PB5        EQU   *
C3A0 BD   CD1B                 JSR   PUTCHR      output a space
C3A3 5A                        DECB              until all done
C3A4 26   FA                   BNE   PB5

C3A6 35   04                   PULS  B           get back # spaces
C3A8 EB   E0                   ADDB  0,S+        add to cur column ctr
C3AA 20   B8                   BRA   PB1         and continue

          C3AC      PB6        EQU   *
C3AC 4D                        TSTA              spc compression byte?
C3AD 2C   00                   BGE   PB4         branch if positive

                    * The char represents a DOS space compression
                    * byte, and indicates the 2's complement of
                    * the actual number of spaces that should
                    * appear at this point. For example, 2
                    * spaces is represented as -2 (X'FE').
                    * Expand this to the proper number of spaces.
                    *
C3AF 40                        NEGA              convert to +ve amt
C3B0 1F   89                   TFR   A,B         transfer to acc B
C3B2 86   20                   LDA   #SP
          C3B4      PB7        EQU   *
C3B4 BD   CD1B                 JSR   PUTCHR      output a space
C3B7 5A                        DECB

C3B8 26   FA                   BNE   PB7
C3BA 20   AB                   BRA   PB1         get another character

          C3BC      PB4        EQU   *
C3BC BD   CD1B                 JSR   PUTCHR      output a char as is
C3BF 5C                        INCB              incr col ctr
C3C0 20   A5                   BRA   PB1         continue
          C3C2      PB3        EQU   *
C3C2 35   36                   PULS  A,B,X,Y     restore regs
C3C4 39                        RTS               and return

                    *
                    * Name     - GETBLK
                    * Function - This routine takes in a trk/sector
                    *            value in acc D, and reads that sector
                    *            into BUFFER.
                    *
                    *            On exit, acc D contains the next
                    *            trk/sec to be read (or 90000 if none)
                    *
                    *            All registers are preserved
                    *
          C3C5      GETBLK     EQU   *
C3C5 34   30                   PSHS  X,Y         save registers

C3C7 8E   C840                 LDX   #SYSFCB     point to an FCB
C3CA 8D   19                   BSR   READSS      read a sector
C3CC 1026 FDAF                 LBNE  ERR02       branch if error

                    *
                    * Transfer bytes just read, to BUFFER
                    *
```

TO BE CONTINUED

DISK SPEED MEASUREMENT

DISK SPEED MEASUREMENT

Mike Jeffreys
787 Continental Circle, 1213
Mt View, CA 94040

Have you ever been unable to read a file written on one drive, but readable on another drive? If so, the problem may be inaccurate drive speed. The following utility allows a user to make a periodic check of floppy disk drive speed. This utility, known as DISKSPD.CMD, works by selecting a drive and making ten separate measurements of the time necessary for the disk to complete one rotation. These measurements are stored in memory until the tenth measurement has been completed, and then a simple plot is produced on the user's terminal. A typical usage of DISKSPD.CMD looks like this:

```
+++DISKSPD,1

SLOW                    CORRECT                    FAST
-51                       V                        +51
                          *
                          *
                           *
                          *
                          *
                          *
                          *
                           *
                          *
                          *
```

If the number of the drive to be tested is not entered on the command line, this program will prompt the user for the required input. Before taking the time to enter this utility into your own system, please make note of the following fact. This utility accesses the disk controller directly in order to measure the time for a single rotation of the disk. Therefore, it may require some modifications for some systems. My system is set up like most SWTPC systems, i.e. with the Disk

Controller board in slots 5 & 6, or memory locations $E814 through $E818. If your system has a disk controller board at the same location using the popular 1771 or 179x series of chips, then this utility will probably run as is.

```
                        *
                        * DISK SPEED CHECK UTILITY
                        *
                        * THIS UTILITY COMMAND CHECKS THE MOTOR SPEED
                        * OF THE SPECIFIED DRIVE.  A 5 INCH DISK
                        * MUST BE IN THE SPECIFIED DRIVE.  NO WRITING
                        * TO THE DISK IS PERFORMED.  A SOFTWARE TIMING
                        * LOOP IS USED; THEREFORE, TIMING ROUTINES MUST
                        * BE CHANGED ON NON 1 MHZ SYSTEMS.
                        *
                        * SAMPLE USAGE:
                        *    ***DISKSPD,1    (CHECKS SPEED OF DRIVE #1)
                        *
                        * WRITTEN BY:  MIKE JOHNSNOY
                        *              707 CONTINENTAL CIRCLE, #1213
                        *              MT VIEW, CA  94040
                        *
                        * BASED ON A ROUTINE WRITTEN BY JOSEPH AULTCHND
                        * PUBLISHED IN '68 MICRO JOURNAL' (FEB 1984)
                        *
                        * LAST MODIFIED:  14 FEB 84
                        *

                        * ND179x EQUATE - NOTE: USER MUST MATCH
                        * THIS POINT TO HIS OWN SYSTEM.  THIS IS
                        * NOT A STANDARD FLEX MEMORY POINT

                E818    COMREG  EQU   $E818     179x COMMAND REGISTER

                        * FLEX EQUATES

                DE09    RESTOR  EQU   $DE09
                CC03    TTYEOL  EQU   $CC03
                CD27    MATCH   EQU   $CD27
                CD15    GETCHR  EQU   $CD15
                CD1E    PSTRNG  EQU   $CD1E
                CD18    PUTCHR  EQU   $CD18
                CD24    PCRLF   EQU   $CD24
                CC03    WARMS   EQU   $CD03

C100                    ORG     $C100   FLEX UTILITY SPACE

C100 20  17     DSKSPD  BRA     START

C102 01         VN      FCB     1       VERSION NUMBER

C103            DRVNUM  RMB     1       DRIVE NUMBER TO TEST
C104            TIME    RMB     20      SPACE FOR 10 TIME VALUES
C118            TSTCNT  RMB     1       NUMBER OF TIMES TESTED

                * GET DRIVE NUMBER FROM COMMAND LINE OR
                * PROMPT USER IF NOT THERE
                *
C119 B9  CD27   START   JSR     MATCH   GET DRIVE NUMBER FROM COMMAND LINE
C11C B1  CC03           CMPA    TTYEOL  WAS IT MISSING ON COMMAND LINE?
C11F 27  0E             BEQ     PROMPT
C121 B1  0D             CMPA    #$0D    WAS IT MISSING ON COMMAND LINE?
C123 27  0A             BEQ     PROMPT
C125 80  30             SUBA    #$30    CONVERT ASCII TO BINARY
C127 B1  01             CMPA    #1      MUST BE DRIVE NUMBER 0 OR 1
)C129 102E 003A         LBGT    ERROR   NO SUCH DRIVE
C120 20  0F             BRA     SELECT  DRIVE OK. GO SELECT IT

C12F 8E  C140   PROMPT  LDX     #PRMSG  POINT AT PROMPT MESSAGE
C132 BD  CD1E           JSR     PSTRNG
C135 BD  CD15           JSR     GETCHR  GET RESPONSE
C138 80  30             SUBA    #$30    CONVERT ASCII TO BINARY
C13A B1  01             CMPA    #1      MUST BE DRIVE NUMBER 0 OR 1
C13C 2E  29             BGT     ERROR   NO SUCH DRIVE

                * SELECT THE SPECIFIED DRIVE
                *
C13E B7  C103   SELECT  STA     DRVNUM  SELECT DRIVE
C141 BE  C103           LDX     #DRVNUM GET READY FOR CALL TO FLEX
C144 30  1D             LEAX    -3,X
C146 BD  DE09           JSR     RESTOR
C149 20  5A             BRA     TIMER

C14B 44 52 49 56  PRMSG  FCC    'DRIVE NUMBER TO BE TESTED? ',4
C14F 45 20 4E 55
C153 4D 42 45 52
C157 20 54 4F 20
C15B 42 45 20 54
C15F 45 53 54 45
C163 44 3F 20 04

C167 8E  C170   ERROR   LDX     #ERRMSG
C16A BD  CD1E           JSR     PSTRNG  PRINT IT
C16D 7E  CD03           JMP     WARMS   RETURN TO FLEX

C170 49 4E 56 41  ERRMSG FCC    'INVALID DRIVE NUMBER',4
C174 4C 49 44 20
C178 44 52 49 56
C17C 45 20 4E 55
C180 4D 42 45 52
C184 04

                * LOOP THROUGH TIMING ROUTINE 10 TIMES, STORING
                * RESULTS IN TIME TABLE FOR DISPLAY LATER.
                *
C185 CA  0A     TIMER   LDB     #10     DO TEN TIMING TESTS
C187 8E  C104           LDX     #TIME   POINT AT START OF TIME TABLE

                * FIND FIRST INDEX PULSE
                *
C18A 1A  10             SEI             DISABLE INTERRUPTS FOR TIMING LOOP
C18C 86  D0             LDA     #$D0    179x 'FORCE INTERRUPT' COMMAND
C18E B7  E818           STA     COMREG  INSURES 'TYPE 1' STATUS
)C191 17  002D          LBSR    DELAY   WAIT 32 uSEC FOR 179x

C194 108E 0000  WAIT    LDY     #$0000  CLEAR TIMING REGISTER
C198 B6  E818           LDA     COMREG  LOOK FOR INDEX PULSE
C19B 84  02             ANDA    #$02    IN BIT 1 OF COMREG
C19D 27  F5             BEQ     WAIT    LOOP UNTIL PULSE DETECTED

                * TIMING LOOP BEGINS TO INCREMENT AFTER THE
                * INDEX PULSE IS DETECTED, THROUGHOUT THE
                * INTERVAL BETWEEN PULSES, AND STOPS WHEN
                * THE INDEX PULSE IS DETECTED AGAIN.
                *
C19F 31  21     CNTHI   LEAY    1,Y     INCREMENT TIMING REGISTER WHILE
C1A1 21  FC             BRN     CNTHI   STILL IN INDEX PULSE
C1A3 12                 NOP
C1A4 B6  E818           LDA     COMREG  STILL IN INDEX PULSE?
C1A7 84  02             ANDA    #$02
C1A9 26  F4             BNE     CNTHI   YES, CONTINUE TO INCREMENT TIMER

C1AB 31  21     CNTLOW  LEAY    1,Y     INCREMENT TIMER UNTIL NEXT
C1AD 21  FC             BRN     CNTLOW  INDEX PULSE DETECTED
C1AF 12                 NOP
C1B0 B6  E818           LDA     COMREG  STILL WAITING FOR INDEX PULSE?
C1B3 84  02             ANDA    #$02
C1B5 27  F4             BEQ     CNTLOW  YES, CONTINUE TO INCREMENT TIMER

                * STORE RESULT OF TEST
                *
C1B7 10AF 01            STY     0,X++   STORE RESULT IN TIME TABLE
C1BA 5A                 DECB            DONE 10 TESTS YET?
C1BB 26  D7             BNE     WAIT    NO, DO ANOTHER TEST
C1BD 1C  EF             CLI             ALLOW INTERRUPTS AGAIN
C1BF 20  07             BRA     GRAPH   GO GRAPH RESULTS

)C1C1 17  0004  DELAY   LBSR    DEL1    DELAY ROUTINE
)C1C4 17  0000  DEL1    LBSR    DEL
 C1C7 39        DEL     RTS

                * TAKE RESULTS OF 10 TESTS NOW STORED IN TIME
                * TABLE AND GRAPH RESULTS ON SCREEN.  A PERFECTLY
                * ACCURATE DRIVE WILL YIELD A COUNT OF DEC 10000
                * (HEX 2710).  THAT VALUE WOULD RESULT IN A POINT
                * BEING PLOTTED IN THE CENTER OF THE GRAPH.
                * A LOWER VALUE REPRESENTS A FASTER DRIVE AND
                * VICE VERSA.  THE GRAPH CONSISTS OF 51 'BINS',
                * RANGING FROM APPX 5% SLOW TO 5% FAST.
                * THE CENTER BIN WILL HOLD POINTS RANGING FROM
                * 9990 TO 10010, WITH 25 EQUAL SIZED BINS
                * (REPRESENTING .2% INCREMENTS) ON EITHER SIDE
                * OF CENTER.
                *
C1C8 8E  C215   GRAPH   LDX     #HEADER PRINT HEADER
C1CB BD  CD1E           JSR     PSTRNG

C1CE C6  0A             LDB     #10     PLOT 10 TIME DATA POINTS
C1D0 F7  C118           STB     TSTCNT
C1D3 8E  C104           LDX     #TIME   X WILL BE POINTER INTO TIME TABLE

                * CHECK FOR DATA POINT OUTSIDE RANGE OF GRAPH
                *
C1D6 108E 2512  RESULT  LDY     #9490   SEE IF MORE THAN 5% FAST
C1DA 10AC 84            CMPY    0,X
C1DD 102E 00AC          LBGT    FAST
C1E1 108E 290E          LDY     #10510  SEE IF MORE THAN 5% SLOW
C1E5 10AC 84            CMPY    0,X
C1E8 1028 0096          LBLT    SLOW

                * DATA POINT NOT TOO HIGH OR TOO LOW.  SO
                * PLOT IT.
                *
C1EC 108E 290E          LDY     #10510  EDGE OF SLOWEST CELL
C1F0 31  A8 EC  PLOT    LEAY    -20,Y   20 IS SIZE OF CELL
C1F3 10AC 84            CMPY    0,X     IN THIS CELL?
C1F6 2F  07             BLE     PLTPNT  YES, GO PLOT POINT
C1F8 86  20             LDA     #$20    NO, SO SKIP TO NEXT CELL
C1FA BD  CD18           JSR     PUTCHR  PRINT THE SPACE
C1FD 20  FE             BRA     PLOT
C1FF 86  2A     PLTPNT  LDA     #'*     PLOT A DATA POINT
C201 BD  CD18           JSR     PUTCHR
C204 BD  CD24           JSR     PCRLF
C207 30  02     NXTPNT  LEAX    2,X     BUMP TABLE POINTER TO NEXT ENTRY
C209 F6  C118           LDB     TSTCNT  CHECK TO SEE IF 10 POINTS PLOTTED
C20C 5A                 DECB
C20D F7  C118           STB     TSTCNT
C210 26  C4             BNE     RESULT

C212 7E  CD03           JMP     WARMS

C215 0D 0A      HEADER  FCC     $0D,$0A
C217 53 4C 4F 57        FCC     'SLOW                   CORRECT'
C21B 20 20 20 20
C21F 20 20 20 20
C223 20 20 20 20
C227 20 20 20 20
C22B 20 20 43 4F
C22F 52 52 45 43
C233 54
C234 20 20 20 20        FDC     '                       FAST',$0D,$0A
C238 20 20 20 20
C23C 20 20 20 20
C240 20 20 20 20
C244 20 20 46 41
```

```
C248 53 54 0D 0A
C24C 20 35 25 20          FCC   '-5%                    V'
C250 20 20 20 20
C254 20 20 20 20
C258 20 20 20 20
C25C 20 20 20 20
C260 20 20 20 20
C264 20 56
C266 20 20 20 20          FCC   '                       +52'.$0D.$04
C26A 20 20 20 20
C26E 20 20 20 20
C272 20 20 20 20
C276 20 20 20 20
C27A 20 20 20 35
C27E 25 0D 0A
C281 04                   FCC   4

C282 86  3C       SLOW    LDA   #'<
C284 BD  CD18             JSR   PUTCHR    PRINT IT
C287 BD  CD24             JSR   PCRLF
C28A 16  FF7A             LBRA  NXTPNT

C28D C6  32       FAST    LDB   #50       SPACE OVER 50 SPACES
C28F 86  20       SKIP    LDA   #$20
C291 BD  CD18             JSR   PUTCHR    PRINT A SPACE
C294 5A                   DECB            DONE YET?
C295 26  F8               BNE   SKIP      NO, PRINT ANOTHER SPACE
C297 86  3E               LDA   #'>
C299 BD  CD18             JSR   PUTCHR
C29C BD  CD24             JSR   PCRLF
C29F 16  FF65             LBRA  NXTPNT

                          END   DSKSPD
```

0 ERROR(S) DETECTED

PC

Enclosed please find a program I call "PC" which is a printer driver that prints in multicolumn form. I know that we've seen an awful lot of printer drivers lately, but I think that this one provides something not yet offered.

I do a lot of long listings -- most often LISTs of committee reports and ASMBlies of programs. I wrote this driver to conserve paper. Basically, it is used like "P", but it intercepts and stores all output until it has enough to fill 2 pages, then prints it all out in 2 column format on one page. It assumes the availability of a 132 column printer -- I use an Epson MX-80 in the condensed mode. I have found an additional benefit of this program is that it helps me to comprehend long assembly listings because you get to see more of the listing at one time. You use the program as shown;

```
+++pc,list myfile              (or)
+++pc/assembly of program PC,asmb pc,pc.cmd
```

The first form will cause the paginated output with a page number and the date at the top of each page. The second form works the same, but adds a title ("assembly of program PC" in this case) to the top of each page. Note that when you use LIST, you should not tell it (LIST) to do paging, because this will interfere with the paging from PC. Note also that PC will cut off all output beyond column 64 of any listing -- you may want to keep a normal listing of programs for your final documentation so that you can read your comments.

For the most part, the accompanying listing should be self-explanatory (at least I still understand it several months later!). Some quick comments are in order however.

The printer size setup characters are defined by equates near the beginning of the program ("NARROW equ 15", "NORMAL equ 18"). The ones shown here work on my Epson, and allow me to have the headings in normal width while the body of the text shows as condensed. If you have a different printer you will need to use its setup characters. If your platen is 14" wide, then you could just set these both to NULL and have everything come out in standard width.

My program lives at $F000 because I have free ram there -- you may want to put it somewhere else. Note though that the driver cannot live in the utility area because it has to stay around while the "other" program executes. Note also that PC grabs a lot of ram under MEMEND to use for its buffer.

The routine at MYPOUT is what actually does the work. It replaces FLEX's normal OUTCH routine. Mostly, it stores incoming characters in the text buffer until it becomes full, then calls routine PRNTPG to actually print the buffer. MYPOUT checks incoming characters to see if they are CR or FF and adjusts the buffer pointer accordingly. For CR characters, it counts the line and advances the pointer to the next "line" position. If the line it terminates would have been the bottom one on a page, it sets flags to indicate a column change. If it had been storing on the right-side column then it calls PRNTPG. Incoming FF characters also cause a column change (and PRNTPG if needed).

Most of the other stuff is pretty obvious. Routine PRNTPG does the actual printout of a page, calling routine TOPAGE to print a tearline and header. TOPAGE calls routines PRPAGE (advance page count and print out the number), and PRDATE which prints the date from FLEX's date registers. Routine INITPG initializes the paging buffer and most of its pointers. The buffer is always initialized to hold all spaces ($20) so that pointer moves within the buffer (i.e. from CR or FF) don't need to worry about the characters they pass over. Location LDATA always points to the start of the left column image, RDATA points to the start of the right column image. BEGLIN points to the start of the current line (this is advanced by LINWID=64 after each CR) and CURPNT points to the location which will receive the next incoming character. Locations LCNTL and LCNTR count lines on the left and right columns respectively (LNCOL=60 determines how many go into a column). CHRCNT counts characters stored in the current line (it is initialized to LINWID=64 after each CR and counts down with each incoming character -- when it reaches zero we ignore characters until a CR or FF). Finally, LRFLAG indicates which column is currently being filled (0=left, 1=right).

You may add this file to your bulletin board system for those who have access to it. For those who don't have modems, I will supply this and a couple other similar and slightly useful programs on an 8" disk for $10.

Just for background, I have been using 68XX computers at work since sometime around the dawn of recorded history, running Motorola's MDOS. About 2 years ago I built my own home machine using a 6809 processor and have been running FLEX on it for most of that time. I use 64K of MCM6665s controlled by an MC6883 SAM chip. My disk controller is based on a Western Digital WD1793 running programmed I/O which was a mistake because I cannot run double density disks in that mode. Someday I will probably build a DMA controller to fix this. I use a Radio Shack TRS-80 Model I as a terminal -- which job it does rather poorly, and it will be replaced whenever I can afford it. I am a EE and was introduced to computers from the electronics end, but seem to be doing more software than hardware of late. I am also a licensed amateur radio operator (WB8KDU), but haven't been active in that field for quite a while.

Hope you like the program !

Sincerely,

Don Korte

Donald N. Korte
2714 Thomas
Flint MI 48504
(313) 238 6716

```
                        *
                        * "PC" utility command
                        *
                        * written by: Don Korte
                        *             2714 Thomas
                        *             Flint, MI  48504
                        * released:   16 JUL 1983
                        * for:        6809 FLEX
                        *
                        * The PC command causes all standard output
                        *  for a specified progra
                        *  to be directed to the system printer.  D
                        *  is paged, titled, and the body is printe
                        *  in a two-column format.  Upon completion
                        *  of the specified progra, a page feed is
                        *
                        *
                        * This program is  modeled on
                        *  the TSC "P" program which is printed
                        *  in their ADVANCED PROGRAMMER'S GUIDE, p.
                        *
                        * Note that this progra may not live in th
                        * standard FLEX utility area because it mus
                        * coresident with the progra it is envelop
                        * (this is because the printer driver lives
                        *  WITHIN this progra).  This is different
                        * the TSC "P" example progra which only ch
                        * the output vector then disappeared.  This
                        * is assembled to live at $F000 which is av
                        * in my syste -- if this isn't possible, i
                        * be put under FLEX if the MEMTOP vector is
                        *
                        * rev 1 7/10/83 dnk
                        * rev 2 7/11/83 moved to F000, paging capab
                        * rev 3 7/12/83 title capability added
                        * rev 4 7/16/83 FCB load moved in runtime
                        *
                              opt    con
                        *
CB40                    FCB    equ    $CB40
CD38                    LOAD   equ    $CD38
D406                    FMS    equ    $D406
D403                    FMSCLS equ    $D403
CD06                    RENTER equ    $CD06
0004                    XFER   equ    $4
CC09                    PAUSE  equ    $CC09
CD1E                    PSTRNG equ    $CD1E
CD3F                    RPTERR equ    $CD3F
CD03                    WARMS  equ    $CD03
CC11                    LSTTRM equ    $CC11
CC02                    EOL    equ    $CC02
CCC0                    PINIT  equ    $CCC0
CCE4                    POUT   equ    $CCE4
CD0F                    OUTCH  equ    $CD0F
CCFC                    PRI    equ    $CCFC
CD4B                    DOCMND equ    $CD4B
CD39                    OUTDEC equ    $CD39
CD27                    NXTCH  equ    $CD27
CC28                    MEMEND equ    $CC28    end of user memory
CC0E                    SYSDAT equ    $CC0E    mo,day,yr
                        *
000C                    FMFEED equ    $0C      form feed char
000D                    NEWLINE equ   $0D      car retn char
0020                    SPACE  equ    $20      space character
003C                    LNCOL  equ    60       # lines per column
0040                    LNWID  equ    64
0F00                    COLSIZ equ    LNCOL*LNWID
1E00                    MEMSIZ equ    2*COLSIZ
                        *
000F                    NARROW equ    15       makes condensed fo
0012                    NORMAL equ    18       makes normal font
                        *
F000                           org    $F000    (out of hare's way
F000 20  01             pc     bra    p1       branch around temp
F002 04                 vn     fcb    4        version number
F003 B6  CCFC           p1     lda    PRI      check syste proce
F006 27  0A                    beq    p12      is it busy ??
F008 BE  CB40                  ldx    #FCB     yes--error point t
F00B C6  10                    ldb    #27      set busy error
F00D E7  01                    stb    1,x      stuff in fcb
F00F 16  0247                  lbra   err3     report error
F012 B6  CC11           p12    lda    LSTTRM   get last terminato
F015 81  0D                    cmpa   #0Dd     is it a cr?
F017 1027 0247                 lbeq   err8     yes--err
F01B 81  CC02                  cmpa   EOL      is it eol char?
F01E 1027 0240                 lbeq   err8     yes--err
F022 BE  F2F0                  ldx    #title   (here we allow for
F025 81  2F                    cmpa   #'/      want title ???
F027 26  0E                    bne    p13      nope
F029 C6  4F                    ldb    #79      max length
F02B BD  CD27           p121   jsr    NXTCH    get char from cod
F02E 81  2C                    cmpa   #',      end of title ??
F030 27  05                    beq    p13      yep
F032 A7  88                    sta    ,x+      no-so store char
F034 5A                        decb            max len yet?
F035 26  F4                    bne    p121     no--okay to keep l
F037 B6  04             p13    lda    #4       store an EOT
F039 A7  84                    sta    0,x      store the EOT
F03B 7F  CC09                  clr    PAUSE    disable pause feat
F03E B6  CCE4                  lda    POUT     examine printer dr
F041 81  39                    cmpa   #$39     is is loaded?
F043 26  26                    bne    p15      if not rts(39) its
F045 BE  F350                  ldx    #forfcb  here we need to lo
F048 108E CB43                 ldy    #FCB+3   copy in "PRINT.SYS
F04C C6  0C                    ldb    #12      twelve bytes worth
F04E A6  80             p14    lda    ,x+      read from mem
F050 A7  A0                    sta    ,y+      write to FCB
F052 5A                        decb
F053 26  F9                    bne    p14      continue til done.
F055 BE  CB40                  ldx    #FCB     now prepare to loa
F058 86  01                    lda    #1       open file for read
F05A A7  84                    sta    0,x
F05C BD  D406                  jsr    FMS      do it...
F05F 1026 01E7                 lbne   err2     branch if error
F063 86  FF                    lda    #$FF     set for binary rea
F065 A7  88 38                 sta    59,x     set compression fl
F068 BD  CB38                  jsr    LOAD     call FLEX's loader
F06B BD  CCC0           p15    jsr    PINIT    initialize port
                        ****************************************
```

```
                        * now we get into the special stuff
                        *  we setup printer, then call
                        *  for user's prograe, then
                        *  clean up.
                        ************************************
F06E BE  CC28           ldx   MEMEND    find normal end of
F071 BF  F342           stx   oldend    and save it
F074 30  89 E200        leax  -MEMSIZ,x form new end of us
F078 BF  CC28           stx   MEMEND    and tell FLEX
F07B 30  01             leax  1,x       first usable for p
F07D BF  F344           stx   ldata     pointer to left co
F080 30  89 0F00        leax  COLSIZ,x
F084 BF  F346           stx   rdata     pntr to right colu
F087 8E  F0B5           ldx   #oypout   get addr of output
F08A BF  CD10           stx   OUTCH+1   replace crt driver
F08D 8E  0000           ldx   #0
F090 BF  F2EE           stx   page      init page num to 0
F093 17  008D           lbsr  initpg    initialize a "page
F096 8D  CD4B           jsr   DOCMND    process command
F099 BD  F1B9           jsr   prntpg    print final page
F09C 8E  CCE4           ldx   #PDUT     temporarily make p
F09F BF  CD10           stx   OUTCH+1   the output device
F0A2 86  12             lda   #NORMAL   this char makes pr
F0A4 BD  CD0F           jsr   OUTCH     so send it !
F0A7 86  0C             lda   #FMFEED   does a top-of-form
F0A9 BD  CD0F           jsr   OUTCH     so send it !
F0AC BE  F342           ldx   oldend    get original MEMEN
F0AF BF  CC28           stx   MEMEND    and put it back
F0B2 7E  CD03           jmp   WARMS     then quit !!!!!
                        ************************************
                        * support routines
                        ************************************
                        *
                        * routine oypout is resident printer driver
                        *
F0B5 34  36     oypout  pshs  x,y,d
F0B7 81  0D             cmpa  #NULINE   end of "line" ?
F0B9 27  1E             beq   clfcr     yes
F0BB 81  0C             cmpa  #FMFEED   top of "page" ?
F0BD 27  74             beq   cfmfd
F0BF 81  20             cmpa  #$20      check for normal p
F0C1 25  14             blo   oypret    too small
F0C3 81  80             cmpa  #$80      or...
F0C5 24  10             bhs   oypret    too big
F0C7 7D  F34B           tst   chrcnt    ok. room on cur li
F0CA 27  0B             beq   oypret    nope--ignore char
F0CC 0E  F34E           ldx   curpnt
F0CF A7  80             sta   ,x+       store char
F0D1 BF  F34E           stx   curpnt    and advanced pntr
F0D4 7A  F34B           dec   chrcnt    and count the char
F0D7 35  86     oypret  puls  x,y,d,pc
                        ***
                        *** newline processing
                        ***
F0D9 7D  F34B   clfcr   tst   lrflag    left or right colu
F0DC 26  30             bne   rlfcr     right !
                        * here if newline when currently
                        *  storing in left column.  Advance
                        *  pointer, count line, indicate
                        *  change to right column if needed.
F0DE F6  F349           ldb   lncntl    check line counter
F0E1 C1  01             cmpb  #1        @ bottom ?
F0E3 27  16             beq   llfcre    yep!
F0E5 7A  F349           dec   lncntl    not bot. cnt this
F0E8 BE  F34C           ldx   beglin
F0EB 30  88 40          leax  LNWID,x   point to next line
F0EE BF  F34C           stx   beglin
F0F1 BF  F34E           stx   curpnt
F0F4 86  40             lda   #LNWID
F0F6 B7  F34B           sta   chrcnt    iniz char counter
F0F9 35  86             puls  x,y,d,pc
                        * at bottom of left column when
                        *  processing newline.
                        *  Must advance to right column.
F0FB BE  F346   llfcre  ldx   rdata     start into right c
F0FE BF  F34C           stx   beglin
F101 BF  F34E           stx   curpnt
F104 86  40             lda   #LNWID
F106 B7  F34B           sta   chrcnt    iniz char counter
F109 7C  F34B           inc   lrflag    say we're on right
F10C 35  86             puls  x,y,d,pc
                        * here if newline when currently
                        *  storing in right column.
                        *  Advance pointer, count line,
                        *  and print page if at bottom.
F10E F6  F34A   rlfcr   ldb   lncntr    check line counter
F111 C1  01             cmpb  #1        @ bottom ?
F113 27  16             beq   rlfcre    yep--now print pag
F115 7A  F34A           dec   lncntr    not bot. cnt this
F118 BE  F34C           ldx   beglin
F11B 30  88 40          leax  LNWID,x   advance to next li
F11E BF  F34C           stx   beglin
F121 BF  F34E           stx   curpnt
F124 86  40             lda   #LNWID
F126 B7  F34B           sta   chrcnt    iniz char counter
F129 35  86             puls  x,y,d,pc
                        * at bottom of right column during
                        *  newline processing.  Print it !
>F12B 17  0058  rlfcre  lbsr  prntpg    really print it
>F12E 17  0022          lbsr  initpg    then iniz for a ne
F131 35  86             puls  x,y,d,pc
                        ***
                        *** process a formfeed
                        *** if on left col, move to right --
                        *** if on right col, print it !
                        ***
F133 7D  F34B   cfmfd   tst   lrflag    left or right colu
F136 27  13             beq   rfmfd     right !
F138 BE  F346           ldx   rdata     left !
F13B BF  F34C           stx   beglin    (so point to right
F13E BF  F34E           stx   curpnt
F141 86  40             lda   #LNWID
F143 B7  F34B           sta   chrcnt    iniz char counter
F146 7C  F34B           inc   lrflag    say we're on right
F149 35  86             puls  x,y,d,pc
>F14B 17  0038  rfmfd   lbsr  prntpg    really print the p
>F14E 17  0002          lbsr  initpg    then inlz for anot
F151 35  86             puls  x,y,d,pc
                        *
                        * routine initpg initializes all pointers
```

```
                        * for a new page, and fills the page with
                        *
F153 C6  20     initpg  ldb   #SPACE
F155 108E 0F00          ldy   #COLSIZ
F159 BE  F344           ldx   ldata     point to left colu
F15C E7  80     init01  stb   0,x+      store space
F15E 31  3F             leay  -1,y
F160 26  FA             bne   init01
F162 108E 0F00          ldy   #COLSIZ
F166 BE  F346           ldx   rdata     point to right col
F169 E7  80     init02  stb   0,x+      store spaces
F16B 31  3F             leay  -1,y
F16D 26  FA             bne   init02
F16F BE  F344           ldx   ldata
F172 BF  F34C           stx   beglin
F175 BF  F34E           stx   curpnt    pointers ready
F178 86  3C             lda   #LNCOL
F17A B7  F349           sta   lncnt1    line counter--left
F17D B7  F34A           sta   lncntr    line counter--righ
F180 86  40             lda   #LNWID
F182 B7  F34B           sta   chrcnt    character counter
F185 7F  F34B           clr   lrflag    say we're on left
F188 39                 rts
                        *
                        * routine prntpg physically prints
                        *  the stored page image.  While printing,
                        *   reg x points into left column,
                        *   reg y points into right column,
                        *   reg b counts lines
                        *   loc chrcnt counts chars in a column
                        *
F189 8D  40     prntpg  bsr   topage    print header junqu
F18B BE  F344           ldx   ldata     left col
F18E 108E F346          ldy   rdata     right col
F192 C6  3C             ldb   #LNCOL
F194 86  40     ppg     lda   #LNWID
F196 B7  F34B           sta   chrcnt    character counter
F199 A6  80     ppg1    lda   ,x+       get char for left
F19B BD  CCE4           jsr   POUT      and print it
F19E 7A  F34B           dec   chrcnt    end of "line" ?
F1A1 26  F6             bne   ppg1      not yet
F1A3 86  20             lda   #SPACE    yep--
F1A5 BD  CCE4           jsr   POUT      so print
F1A8 BD  CCE4           jsr   POUT      some spaces
F1AB BD  CCE4           jsr   POUT      for beauty
F1AE B6  40             lda   #LNWID
F1B0 B7  F34B           sta   chrcnt    reinit character c
F1B3 A6  A0     ppg2    lda   ,y+       get char for right
F1B5 BD  CCE4           jsr   POUT      and print it
F1B8 7A  F34B           dec   chrcnt    end of line ?
F1BB 26  F6             bne   ppg2      nope
F1BD B6  0D             lda   #NULINE   yep
F1BF BD  CCE4           jsr   POUT      so make it happen
F1C2 5A                 decb            done with page?
F1C3 26  CF             bne   ppg       no--go print anoth
F1C5 86  0C             lda   #FMFEED   yep
F1C7 BD  CCE4           jsr   POUT      so move paper up
F1CA 39                 rts
                        *
                        * routine topage prints tear line, header i
```

```
                        *    and advances page counter
                        *
F1CB 34  36     topage  pshs  x,y,d
F1CD BE  CCE4           ldx   #POUT     restore normal pri
F1D0 BF  CD10           stx   OUTCH+1   so PSTRNG works di
F1D3 8E  F2AE           ldx   #header
F1D6 BD  CD1E           jsr   PSTRNG    " --- "
F1D9 8D  17             bsr   prpage    print page num
F1DB 8D  26             bsr   prdate    print date
F1DD 8E  F2F0           ldx   #title
F1E0 BD  CD1E           jsr   PSTRNG    pprint user's titl
F1E3 BE  F2BA           ldx   #head1
F1E6 BD  CD1E           jsr   PSTRNG    end of it
F1E9 8E  F0B5           ldx   #mypout   now back to interc
F1EC BF  CD10           stx   OUTCH+1   printer driver...
F1EF 35  36             puls  x,y,d
F1F1 39                 rts
                        *
                        * routine PRPAGE advances page count,
                        *   then prints page number
                        *
F1F2 BE  F2EE   prpage  ldx   page
F1F5 30  01             leax  1,x       advance count
F1F7 BF  F2EE           stx   page
F1FA BE  F2EE           ldx   #page     point to counter
F1FD C6  01             ldb   #1        want leading space
F1FF 8D  CD39           jsr   OUTDEC    print number
F202 39                 rts
                        *
                        * routine prdate prints spaces then the dat
                        *
F203 C6  28     prdate  ldb   #40       forty
F205 86  20             lda   #SPACE    spaces
F207 BD  CD0F   prd01   jsr   OUTCH     are
F20A 5A                 decb            printed
F20B 26  FA             bne   prd01     by this !
                        * print date
F20D B6  CC0F           lda   SYSDAT+1  day
F210 B7  F341           sta   num+1     temporary storage
F213 7F  F340           clr   num+0     which is 2 bytes l
F216 8E  F340           ldx   #num      point to it
F219 5F                 clrb            (want spaces suppr
F21A BD  CD39           jsr   OUTDEC
                        * print month
F21D B6  CC0E           lda   SYSDAT+0  month 1-12
F220 4A                 deca            month 0-11
F221 48                 asla            *2
F222 48                 asla            *4
F223 8E  F2BD           ldx   #months   table of month nam
F226 30  86             leax  a,x       point at proper na
F228 C6  04             ldb   #4        length is 4 chars
F22A A6  80     prd02   lda   ,x+       print one of them
F22C BD  CD0F           jsr   OUTCH     (here)
F22F 5A                 decb            done yet?
F230 26  F8             bne   prd02     not yet...
F232 86  20             lda   #SPACE    a space for
F234 BD  CD0F           jsr   OUTCH     the sake of beauty
                        * print year
F237 CC  076C           ldd   #1900     base year
F23A F8  CC1B           addb  SYSDAT+2  year
```

```
F23D 89   00          adca  #0        double precision!
F23F FD   F340        std   num       keep in mem
F242 BE   F340        ldx   #num      point to it
F245 5F               clrb            (spaces suppressed
F246 BD   CD39        jsr   OUTDEC
F249 39               rts
                      ************************************
                      * now error routines
                      ************************************
F24A A6   01    err2  lda   1,x       get error code
F24C 81   04          cmpa  #NFER     is it "no such fil
F24E 26   09          bne   err3      nope
F250 30   8D 0014     leax  nopst,pcr point to message
F254 BD   CD1E  err25 jsr   PSTRNG    print message
F257 20   03          bra   err4
F259 BD   CD3F  err3  jsr   RPTERR    report harder erro
F25C BD   D403  err4  jsr   FMSCLS    close all files
F25F 7E   CD03        jmp   WARMS     return to FLEX
F262 30   8D 0021 err8  leax  erstr,pcr point to string
F266 20   EC          bra   err25     and print it...
                      *
F268 65 72 72 6F nopst fcc   /error: "PRINT.SYS" not foun
F26C 72 3A 20 22
F270 50 52 49 4E
F274 54 2E 53 59
F278 53 22 20 6E
F27C 6F 74 20 66
F280 6F 75 6E 64
F284 20 21
F286 04               fcb   4
F287 75 73 61 67 erstr fcc   "usage: PC,command  or PC/ti
F28B 65 3A 20 50
F28F 43 2C 63 6F
F293 6D 6D 61 6E
F297 64 20 20 6F
F298 72 20 50 43
F29F 2F 74 69 74
F2A3 6C 65 2C 63
F2A7 6F 6D 6D 61
F2AB 6E 64
F2AD 04               fcb   4
F2AE 12 2D 2D 2D header fcc   NORMAL,/---/,MULINE
F2B2 0D
F2B3 50 41 47 45      fcc   /PAGE: /,4
F2B7 3A 20 04
F2BA 0F 0D 04   headl fcb   NARROW,MULINE,4
F2BD 20 4A 41 4E months fcc  / JAN FEB MAR APR MAY JUN/
F2C1 20 46 45 42
F2C5 20 4D 41 52
F2C9 20 41 50 52
F2CD 20 4D 41 59
F2D1 20 4A 55 4E
F2D5 20 4A 55 4C      FCC   / JUL AUG SEP OCT NOV DEC/
F2D9 20 41 55 47
F2DD 20 53 45 50
F2E1 20 4F 43 54
F2E5 20 4E 4F 56
F2E9 20 44 45 43
                      *
F2ED            lncnt rmb   1         line counter (rev
F2EE            page  rmb   2         page counter
F2F0            title rmb   80
F340            num   rmb   2
F342            oldend rmb  2         stores orig MEMEND
F344            ldata rmb   2         -> start of left c
F346            rdata rmb   2         -> start of right
F348            lrflag rmb  1         0=left 1=right
F349            lncntl rmb  1         line count left si
F34A            lncntr rmb  1         line count right s
F34B            chrcnt rmb  1         char count "this"
F34C            beglin rmb  2         -> start current l
F34E            curpnt rmb  2         -> current char
                *
                * the following code is loaded into
                * the system FCB when the PC command is
                * executing (load of pc fails if this
                *  is stored by org in this source)
F350 FF         forfcb fcb  $FF
F351 50 52 49 4E       fcc  /PRINT/
F355 54
F356 00 00 00          fcb  0,0,0
F359 53 59 53          fcc  /SYS/
                *
                *
                       end  pc

0 ERROR(S) DETECTED

SYMBOL TABLE:

COLSIZ 0F00  DOCMND CD48  EOL    CC02  FCB    C040  FNFEED 0
FMS    D406  FMSCLS D403  LNCOL  003C  LNMID  0040  LOAD   C
LSTTRM CC11  MEMEND CC2B  MEMSIZ 1E00  NARROW 000F  NFER   0
NORMAL 0012  NULINE 000D  NXTCH  CD27  OUTCH  CD0F  OUTDEC C
PAUSE  CC09  PINIT  CCC0  POUT   CCE4  PRI    CCFC  PSTRNG C
RENTER CD06  RPTERR CD3F  SPACE  0020  SYSDAT CC0E  WARMS  C
beglin F34C  cfmfd  F133  chrcnt F348  clfcr  F0D9  curpnt F
err2   F24A  err25  F254  err3   F259  err4   F25C  err8   F
erstr  F287  forfcb F350  headl  F2BA  header F2AE  init01 F
init02 F169  initpg F153  ldata  F344  llfcre F0FB  lncnt  F
lncntl F349  lncntr F34A  lrflag F348  months F2BD  oypout F
oypret F0D7  nopst  F268  num    F340  oldend F342  p1     F
p12    F012  p121   F02B  p13    F037  p14    F04E  p15    F
page   F2EE  pc     F000  ppg    F194  ppg1   F199  ppg2   F
prd01  F207  prd02  F22A  prdate F203  prntpg F189  prpage F
rdata  F346  rfmfd  F14B  rlfcr  F19E  rlfcre F12B  title  F
topage F1CB  vn     F002
```

BIT BUCKET

```
8010 * PROGRAM 'CELESTE'. C.O. EAGLE. JANUARY. 10 2.
8020 *
8030 * COMPUTES TOPOCENTRIC DIRECTION OF THE PLANETS
8040 *
8050 * AZIMUTH= ANGLE. IN DEGREES. POSITIVE CLOCKWISE FROM NORTH
8060 * ELEVATION= ANGLE. IN DEGREES. POSITIVE ABOVE THE HORIZON
8070 *
8080 DIM A$(8),L(3),N(7),U(3),A(3,3),R(9,11)
8090 LET RD=PI/180:STRING=71:LIN =100:GOSUB 1010
8100 *
8110 HOME :INPUT "LATITUDE: +NORTH, -SOUTH)";L1
8120 PRINT :INPUT "LONGITUDE(+WEST, -EAST)";L2
8130 PRINT :INPUT "TIME ZONE";Z5
8140 PRINT :INPUT "DATE: MONTH, DAY, YEAR";M5,D5,Y5
8150 PRINT :INPUT "TIME: HOURS, MINUTES, SECONDS";H1,M1,S1
8160 PRINT :INPUT "DAYLIGHT SAVINGS TIME(1=YES, 0=NO)";F1
8170 *
8180 * COMPUTE JULIAN DATE, OBLIQUITY AND SIDEREAL TIME
8190 GOSUB 3010
8200 *
8210 * COMPUTE GEOCENTRIC POSITION OF THE SUN
8220 LET K=9:GOSUB 3010
8230 LET X1=-X2:Y1=-Y2
8240 *
8250 HOME :PRINT "PLANETARY MENU":PRINT
8260 FOR I=1 TO 8:PRINT TAB(10);I;A$(I):NEXT I
8270 PRINT :INPUT "SELECTION";K
8280 *
8290 * COMPUTE GEOCENTRIC POSITION OF PLANET
8300 GOSUB 3010:X3=X1+X2:Y3=Y1+Y2
8310 LET Y4=Y3*C  (E))-Z2*R(N(E)):Z3=Y3*R(N(E))+Z2*C(N(E))
8320 *
8330 * COMPUTE TOPOCENTRIC POSITION OF PLANET
8340 LET R=SQR(X3*X3+Y4*Y4+Z3*Z3)
8350 LET U(1)=X3/R:U(2)=(Y4/R)*U(3)=Z3/R:GOSUB 4010
8360 *
8370 PRINT #7,TAB(5);:"PROGRAM CELESTE":PRINT #7:PRINT #7,TAB(5);A$(K)
8380 PRINT #7:PRINT #7,TAB(5);"LATITUDE";L1;TAB(50);"LONGITUDE";L2
8390 PRINT #7:PRINT #7,TAB(5);"DATE";M5;D5;Y5;TAB(50);"TIME";H1;M1;S1
8400 PRINT #7:IF E<0 THEN PRINT #7,TAB(5);"BELOW THE HORIZON"
8410 PRINT #7:PRINT #7,TAB(5);"AZIMUTH";Q/RD;TAB(50);"ELEVATION";E/RD
8420 *
8430 PRINT :INPUT "ANOTHER DIRECTION(1=YES, 0=NO)";F:IF F=0 END
8440 PRINT :INPUT "ANOTHER LOCATION (1=YES, 0=NO)";F:IF F=1 GOTO 110
8450 PRINT :INPUT "ANOTHER DATE(1=YES, 0=NO)";F:IF F=1 GOTO 150
8460 PRINT :INPUT "ANOTHER TIME(1=YES, 0=NO)";F:IF F=1 GOTO 150
8470 GOTO 230
8480 *
1000 * READ PLANET NAMES AND ORBITAL ELEMENTS DATA SUBROUTINE
1010 FOR I=1 TO 9:FOR J=1 TO 11:READ M(I,J):NEXT J:NEXT I
1020 FOR I=1 TO 8:READ A$(I):NEXT I:RETURN
1030 *
1040 DATA .3870984, .20561421, 2.046E-5, .122223323, 3.2477668E-5
1050 DATA .501847459, .006462615, .022051959, .020685788
1060 DATA 1.70511195, 2608.78753, .7233316, .00602869, -4.774E-5
1070 DATA .059238827, 1.75551103E-5, .049183109, .00806521
1080 DATA 1.32264435, .0157053145, 3.71862418
1090 DATA 1021.32035, 1.5236915, .0933129, 9.2044E-5
1100 DATA .032294409, -1.17809772E-5, 4.98172492, .018678954
1110 DATA .051484037, .013456343, 5.57684053, 334.85355
1120 DATA 5.202561, .04833475, .00016418, .022841754
1130 DATA -9.94158938E-5, 4.76959287, .010462056, 1.73261451
1140 DATA .017637876, 3.93272122, 52.9653474, 9.554747
1150 DATA .05589232, -3.455E-4, .043502671, -6.8397514E-5
1160 DATA 5.98458451, .0189406742, 1.96856365, .0152481382
1170 DATA 3.86246321, 21.3208951, 19.21814
1180 DATA .0463444, -2.658E-5, .0134020382, 1.0913156E-5
1190 DATA 1.7116715, .017284844, 1.28241727, .008703395
1200 DATA 1.2679611, 7.47662597, 38.18957, .08899784
1210 DATA 6.33E-6, .0310536253, -1.6656744E-4, 4.81791101
1220 DATA .005683481, 2.28881997, .019180034, .450524488
1230 DATA 3.812989981, 39.517738, .24864438, 8, .29926728
1240 DATA 0, 1.9813255, 0, 1.98161466, 0, 4.0133377
1250 DATA 2.5292055, 1.00000013, .01675104, -4.18E-5, 0, 0
1260 DATA 1.74663681, .038885264, 0, 0, 0.25038358, 620.301946
1270 *
1280 DATA MERCURY,VENUS,EARTH,MARS,JUPITER,SATURN,URANUS,  PLUTO,PLUTO
1290 *
2000 * JULIAN DATE, OBLIQUITY AND SIDEREAL TIME SUBROUTINE
2010 LET J1=367*Y5-INT(7*(Y5+INT((M5+9)/12))/4)+INT(275*M5/9)+D5+1721013.5
2020 LET T1=(H1+(M1+S1/60)/60+Z5-F1)/24:T2=(J1-2415020)/36525
2030 LET T3=T2*T1/876600:E1=.409319755-2.2711804*E-4*T3
2040 LET S2=1.73993589+628.33195*T2-SGN(L2)*L2*RD
2050 LET S3=S2+.262516163*T1:RETURN
2060 *
3000 * HELIOCENTRIC POSITION FROM ORBITAL ELEMENTS SUBROUTINE
3010 LET N(1)=A(K,1):J=2:FOR I=2 TO 6
3020 LET N(I)=N(K,J)+N(K,J+1)*T3:J=J+2:NEXT I:A=N(6):B=N(2)
3030 LET S4=SIN(A):S5=SIN(2*A):S6=SIN(3*A):S7=SIN(4*A)
3040 LET C=B*(2*S4+B*(.5*S5/4+B*((13*S4-3*S6)/12)+B*((103*S7-44*S5)/96)))
3050 LET N(7)=C+N(6):A=N(1)-N(2)*N(2)/(1+N(2)*COS(N(7)))
3060 B=N(5):C=COS(N(4)+N(7)):S=SIN(N(4)+N(7)):F=N(3)
3070 LET X2=A*(COS(B)*C-SIN(B)*COS(F)*S):T2=A*(SIN(B)*C+COS(B)*COS(F)*S)
3080 LET Z2=A*S*SIN(F):RETURN
3090 *
4000 * EQUATORIAL TO TOPOCENTRIC TRANSFORMATION SUBROUTINE
4010 LET A=SIN(L1*RD):B=COS(L1*RD):C=SIN(S3):D=COS(S3)
4020 LET A(1,1)=A*D:A(1,2)=A*C:A(1,3)=-B
4030 LET A(2,1)=-C:A(2,2)=D:A(2,3)=0
4040 LET A(3,1)=B*D:A(3,2)=B*C:A(3,3)=A
4050 FOR I=1 TO 3:S=0:FOR J=1 TO 3:S=S+A(I,J)*U(J):NEXT J:L(I)=S:NEXT I
4060 LET E=ATAN(L(3)/SQR(1-L(3)*L(3)))
4070 LET A=L(2)/COS(E):B=-L(1)/COS(E)
4080 IF ABS(A)<1E-8 THEN Q=0:RETURN
4090 LET Q=(2-SGN(A))*.5*PI:IF ABS(B)<1E-8 RETURN
4100 LET Q=Q+ CK(A)*SGN(B):A=ABS(ATAN(A/B)):Q=.5*PI):RETURN
```


SUPPORT YOUR ADVERTISERS

C Programmer's Library
$19.95 (B. Dalton price)
Author: Dr. Jack Purdum, Tim Leslie & Alan Stegemoller
A Staff Review

A very recent addition to the how-to of 'C' books is the C Programmer's Library. Published by QUE Corp. And burdensomely corrected by this reviewer, with three looseleaf pages of corrections to, of all things, the source files included.

This is not intended to be a complete review, as is our policy, but more an 'alert' to those of our readers who might contemplate purchase of this book. While instructive in its 'text' portions, the source files (which to many are the most important parts) are rife with errors, omissions, deletions, typo foulups, etc. So much so that it becomes difficult to use them as study material. After spending the better part of an hour correcting the book, I found myself wondering how many other errors are there, but not attended to. So much so that I have set it aside until I feel most of the goofs have been found and rectified for the reader.

For the educator, intent on referring his/her students to solid C material, this is NOT it! Mainly due to the faults mentioned. The source code (corrected) may compile and run, but the distraction of always wondering what other errors lurk within to snare the unwary student, makes for an unsatisfactory learning situation.

Even the manner of correction is stupidly approached. Instead of reprinting each page in error so that it may be attached by clear tape, or what have you, without a lot of fuss, the corrections are noted, one after the other, for 3 full pages (of the four copies of this title on the bookstore shelf, not all had the corrections loosely inserted, one of our editors purchased a personal copy the night before we purchased our review copy, his does not have any correction sheets inserted). Try getting all that in troops, without leaving something out. You did? Well, maybe.

Another very annoying point is the type font selected by either the authors, publisher or both. It is called 'Digital'. It has one especially bad drawback, the character 'l' (ell) and the numeral '1' (one) are so similar that it is practically impossible to tell th apart.

Example:

```
i2bytes = sizeof(struct i2_link) + max_i2 * (sizeof(struct i2_elem) *
        ikeylen - 1);
i3bytes = max_i3 * (sizeof(struct i3_elem) * ikeylen - 1);

temp = sizeof(struct global) * ikeylen - 1 * max_s * (sizeof(
        struct s_elem) * ikeylen - 1) * i2bytes * i3bytes *
        (i2bytes > i3bytes ? i2bytes : i3bytes) * sizeof(
        struct recov) + (r_max - 1) * sizeof(long);
```

Is it one two bytes = ?
Or is it l(ell) two bytes = ?

Page after page of this type of mental jabbing by the font renders it practically useless as tutorial material. Also there is another major fault in this type of font selection.

1. Had they used a plain vanilla dot matrix (or daisy wheel) output, it would have been far more legible, as most computer types spend many hours eyeballing dot matrix fonts (mosty anyone can read a daisy wheel output without much difficulty), the eye and mind understand them. Not all of us are privy to 'Digital' (or want to be)!

2. Also, had they used the dot matrix font it would probably have been spit out by a computer of some sort. One that most likely also compiles C. This being the case (no pun intended), then it would not have contained all the other errors or omissions. Because if an attempt to compile the source first had been made (a logical assumption) then the printed output would have been correct. Or the system would have 'bombed' the compile session, as the source is presented in this book, thereby alerting someone to the fact of omissions and/or errors existing. The results would probably have been a book that I could wholeheartly recommend. As it is, unless you like to cut out things and paste them into places where they may or may not fit easily, and enjoy puzzles of this sort (looking for additional errors, etc.), I would recommend holding off on this one.

FOR THE ELEKTRA

- OS-9™ with Editor, Assembler, and Debugger **$250.00**
- STAR-DOS™ Level 1 (FLEX™ compatible but with up to 10 active drives: (i.e. 4 8" floppies, 4 5" floppies, and 2 Winchester drives) **$75.00**
- STAR-DOS™ Adaptation Guide **$50.00**

FOR OS-9™ by Epstein Associates

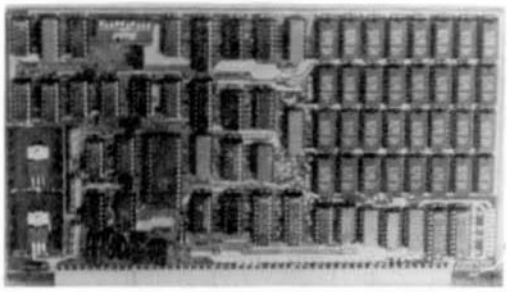
- Super Modem Program with autodial, configuration file, etc. (Available exclusively through AAA Chicago Computer Center) **$100.00**

THE ELEKTRA SUPER FLOPPY CONTROLLER

- Emulates the DC-1, DC-2, DC-3, DC-4 as well as the GIMIX #28, #38, #48, and #58 controllers **$295.00**

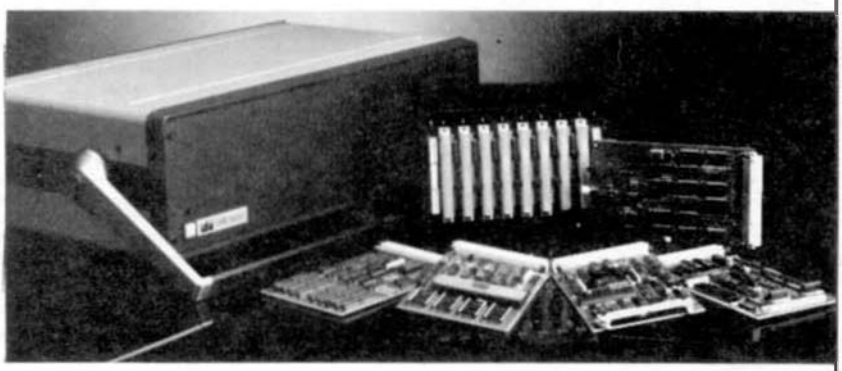
RACKMOUNT ELEKTRA COMPUTER CABINET

- 17" W x 21.5" D x 6.7" H
  Holds two half height 5¼" floppy drives **$250.00**



2MHz 256K Memory Board
with on board DAT by
COMPUTER EXCELLENCE, INC.
$749.00

Now Available
68010
and
VME Bus
by
MIZAR INC.
Phone for details



Phone:
AAA Chicago Computer Center
Technical Consultation available most weekdays from 4 p.m. to 6 p.m. CST
(312) 459-0450 120 Chestnut Lane Wheeling, IL 60090
See our catalog and ordering information on the next page.

ELEKTRA COMPUTER SYSTEM Includes chassis, dual port serial interface with two cables, CPU 8/9, 4K Humbug, 56K static RAM, super floppy controller with inboard ribbon cable, Star-Dos, dual 80 track DSDD floppy drives (other combinations available; phone) $2795.00

ELEKTRA COMPUTER CABINET THE LARGEST SS-50 COMPUTER CABINET AVAILABLE! Made of heavyweight 0.090" thick aluminum. Interior is 18-1/2" wide by 21-7/8" deep by 6-3/4" high. Heavy duty A.C. line cord, A.C. fuse holder, EMI filter. Fan with filter. Back panel has 10 cutouts for 'D' type data connectors. Front panel has key on/off power switch, 2 illuminated push button switches (Reset and NMI/Abort), and two cutouts for 5-1/4" disk drives. $250.00

RACKMOUNT ELEKTRA COMPUTER CABINET 17"w x 21.5"d x 6.7"h. Holds two half height 5-1/4" floppy drives. 250.00

Filler Plate for 5-1/4" drive opening: $10.00 Fan Filter: $10.00

POWER SUPPLY Highest quality linear power supply. CONSERVATIVELY rated at 15a @ 8v, 3a @ 16v, 3a @ -16v. Multi-tapped primary for line tuning. $200.00

DISK REGULATOR BOARD WITH CABLES Standard version for 2 floppy drives $50.00. Heavy duty version for 1 Winchester drive and 1 floppy drive. $75.00

AUXILIARY POWER SUPPLY to power second Winchester drive. $125.00

ELEKTRA UNIVERSAL SS-50/SS-50C MOTHERBOARD Heavyweight 0.125" thick, 18" long by 8" wide. 11 memory (50 pin) slots, 8 I/O (30 pin) slots. Complete address decoding and selection, as well as extended address capability, for I/O slots. Choice of 4, 8, or 16 addresses per I/O slot. 1" spacing between all memory and I/O slots. On board baud rate generator with low and high ranges providing jumper selectable rates of 75 through 38,400 for each of the five baud rate lines, slow device circuitry permitting 1 Mhz 30 pin disk controllers to run with 2MHz 50 pin CPU boards.

Mounting hardware $5.00 Bareboard w/documentation $80.00
Kit w/gold connectors $400.00 Assembled w/gold connectors $460.00
Kit w/pin connectors $320.00 Assembled w/pin connectors $380.00

ELEKTRA CHASSIS Includes cabinet, 110v power supply, power supply cables, standard disk regulator board with power cables, motherboard with gold square pin connectors, assembled and tested. $950.00

ELEKTRA CPU 8/9 Use either the 6802 or 6808 (to run 6800 software) or 6809. Has provision for up to 3 2716 Eproms, 1K scratchpad, MC6840 triple timer, and an optional baud rate generator providing baud rates from 110 through 38,400 baud in two user selectable ranges. Run OS-9™, FLEX™, STAR-DOS.

Bareboard $50.00 Kit $225.00 Assembled $275.00
Optional Baud Rate Generator $25.00

ELEKTRA DPS DUAL PORT SERIAL CARD Fits the standard 30 pin SS-50 bus I/O slot. Can be configured for 4 or 16 addresses per port. RTS, CTS, DTR, DCD, IRQ, FIRQ/NMI, and baud rate can be appropriately implemented for each port.

Bareboard $25.00 Kit $75.00 Assembled $95.00
Cable with jack socket assemblies (two needed per board) Each $25.00

ELEKTRA DPP DUAL PORT PARALLEL CARD Fits the standard 30 pin SS-50 bus I/O slot. Can be configured for 4 or 16 addresses per I/O slot. The direction of the TTL buffers can be controlled by either on board jumper connectors or by a signal from the peripherals. The interrupt request line for each port may be individually turned to either the IRQ or FIRQ/NMI bus line.

Bareboard $25.00 Kit $60.00 Assembled $80.00
Cable with jack socket assemblies (two needed per board) Each $25.00

ELEKTRA 64K STATIC RAM/ROM MEMORY BOARDS with gold connectors (tin available) Assembled and tested. With 56K RAM $269.00 With 64K RAM $299.00
Kit With 56K RAM $219.00 With 64K RAM $249.00

ELEKTRA UNIVERSAL SUPER FLOPPY CONTROLLER THE BEST 30 PIN FLOPPY DISK CONTROLLER THAT YOU CAN BUY! Controls up to four 5-1/4" drives and four 8" drives for a total of eight system drives. (FLEX system limit is four drives). Single density or double density, 1MHz or 2MHz, 6800 or 6809. (Double density 8" must be at 2MHz; all other combinations of performance are possible.) Analog phase locked loop data separators with separate adjustments for 5" and 8" drives. Analog write precompensation circuit with separate adjustments for 5" and 8" drives. Designed to meet the data hold requirements of Western Digital floppy controller IC. Assembled and tested. $295.00

Disk with drivers, setup, and formatting utilities. Specify FLEX 2.0, 6800 Gen FLEX, FLEX 9.0, FLEX 9.1, or 6809 Gen FLEX, 5" or 8" $30.00
Disk with drivers for OS-9 (Specify 5" or 8") $50.00
STAR-DOS (Specify ELEKTRA or DC-2, 5" or 8") $75.00 Adaptation guide $50.00

ELEKTRA WINCHESTER SYSTEMS THE BEST WINCHESTER SYSTEMS THAT YOU CAN BUY! Has automatic error detection and CORRECTION of up to 11 bit burst errors. SS-50 bus, extended addressing capabilities, DMA, on board sector buffer, drivers included for 6809 FLEX or OS-9. Specify whose version of FLEX that you are using. Drivers for FLEX2 (6800) are available for an additional $100.00. Price includes host interface, controller, drive(s), and cables.

7 Megabyte single drive sys. $1895.00 14 Megabyte dual drive sys. $2995.00
12 Megabyte single drive sys. $2295.00 24 Megabyte dual drive sys. $3595.00
19 Megabyte single drive sys. $2995.00 38 Megabyte dual drive sys. $4695.00
(19 Megabyte drives are the largest that can be supported by FLEX)

Circuit boards, cables, software (No drives) 995.00
SS-50C DMA Bus interface board only 695.00

ELEKTRA HD-5 Cabinet for dual 5 1/4" floppy drives with power supply, line cord, fuse, power switch, and power cables to drives. 150.00

ELEKTRA HD-5W As above but with EMI filter, fan, and heavy duty power supply. Powers 1 floppy and 1 Winchester. 199.00

5" ribbon cable for dual outboard 5-1/4" disk drives 40.00
2" ribbon cable for dual inboard 5-1/4" disk drives 35.00
Custom cables available Phone

ELEKTRA HD-8 Dual 8" drive cabinet, EMI filter, fan with filter, power supply and power supply cables. 350.00

8" ribbon cable for dual 8" disk drives 45.00

ELEKTRA 30 PIN PROTOTYPING BOARD 20.00
ELEKTRA 50 PIN PROTOTYPING BOARD 40.00

GOLD 10 PIN CONNECTORS (Specify male with square pins or female) 1.50
TIN 10 PIN CONNECTORS (Specify male with square pins or female) .50

ELEKTRA is a trademark of AAA Chicago Computer Center.
FLEX and UniFLEX are trademarks of Technical Systems Consultants, Inc.
HELIX is a trademark of Hazelwood Computer Systems
OS-9 and BASIC09 are trademarks of Motorola Inc. and Microware Systems Corp.

AAA CHICAGO COMPUTER CENTER (312) 459-0450
120 CHESTNUT LANE • WHEELING, IL 60090
Technical consultation available 4 PM to 6 PM most weekdays. Closed evenings and weekends.

TERMS Minimum order $20.00. Shipping and handling estimates within the Continental U.S., add 3% (MINIMUM $2.50). Illinois residents add 7% sales tax. We will refund your overestimated shipping and handling charges. Foreign shipping and handling, add 10% (MINIMUM $10.00). Foreign orders must be prepaid in U.S. dollars. Checks must be drawn on a U.S. bank. Heavy foreign items will be shipped air freight collect. Please phone between 4 PM and 6 PM weekdays if questions arise regarding shipping fees. Master Charge, Visa, and American Express honored.

Our apology. We are not staffed to answer technical inquiries through the mail. Please phone for technical help during the hours indicated above. The too frequent changing of our inventory and prices makes it uneconomical to publish a catalog. Our ads are intended to serve that purpose. Prices and inventory are subject to change without advance notice.

SUPER MODEM PROGRAM Single character commands. No interrupts required. Transmit manually or transmit disk files (text) of any length to distant computer. Receive and save disk files (text) on local disk system. X-on/X-off supported. Tested for full duplex at speeds up to 9600 baud. Half duplex option. Echo option. Replaces CR with CR/LF (user option). Slow disk file transmit option.

Please specify 6800 or 6809, SSB or FLEX™, 5" or 8".
Instruction Manual and disk with both source and object code $75.00

OS-9 Super Modem Program by Epstein Associates ...ith autodial configuration file, etc. 100.00

ALL IN ONE
Editor — Text Processor — Mailing Labels — Mailing Lists — Multiple Form Letters
Use any CRT terminal and printer -- Best Package For The Money Anywhere!

Specify 6800 or 6809, SSB or FLEX™, 5" or 8" 75.00
Add $35.00 for printed source listing; add $100 for source on disk.
All-In-One, Write'n Spell, and Spell'n Fix package 250.00

Software by Technical Systems Consultants, Inc.	FLEX™ Source (Uell)	FLEX™ Source (Disk)	FLEX™ Man. Only	FLEX™ Object w/Man.	UniFLEX™ Add. Man.	UniFLEX™ Man. Only	UniFLEX™ Object w/Man.
Gen. FLEX w/Edit & ASMB	—	—	—	250	—	—	—
FLEX 9.1 (OC-2) w/Edit & ASMB	—	—	25	150	40	100	550
Advanced Programmers Guide	—	—	25	—	—	—	—
Editor	100	250	25	50	—	—	—
Assembler	150	250	25	50	—	—	—
Debug	175	250	25	75	—	—	—
Extended Basic	—	—	25	100	20	50	200
Basic Precompiler	—	—	25	50	10	25	150
Sort/Merge	—	—	25	75	20	35	150
Utilities	—	Inc	25	75	10	25	150
Diagnostics	—	—	25	75	—	—	—
Text Processor	150	250	25	75	20	35	150
68000 X-ASMB on 6809	—	—	25	250	20	35	300
Pascal	—	—	50	200	25	50	300
Rel ASMB/Linking Loader	—	—	25	150	20	35	175
6800 X-ASMB on 6809	—	—	—	100	—	—	—
Cobol	—	—	—	—	30	75	750
Fortran 77	—	—	—	—	35	65	450

Software by Microware Systems Corp. (Suggested List Prices, varies/mfg)	Run-Time Package	Source	Manual Only	Object w/Man.
OS-9™ Level 1 w/Edit, Asm, Debug		400.00	40.00	250.00
OS-9™ Level 2 w/Edit, Asm, Debug		400.00	40.00	500.00
OS-9™ Edit, Asm, Debug Pkg			25.00	125.00
Device Driver for Disk Controller (Specify Model)		100.00		
Device Driver for ACIA and PIA		50.00		
Clock Driver for 6840 and 6967 clock chips		35.00		
Entertainment Pack I, or File Handler Toolbox, or NineCom			10.00	5.00
Print Spooler (Level 2 only)			15.00	95.00
Virtual Disk Driver (Level 2 only)			10.00	5.00
RMA Relocatable Macro Assembler			20.00	125.00
RMA/68000 Cross Assembler				400.00
BASIC09™ W/Run-Time	50.00	N/A	25.00	200.00
BASIC09 Tutor Guide Book			18.95	
"C" Compiler			25.00	250.00
C Programming Language (Kernighan & Ritchie)			19.95	
CIS Cobol Compiler w/Forms 2 Prog. Gen.	50.00	N/A	40.00	400.00
Pascal Compiler	50.00	N/A	25.00	250.00
Sage Application Generator	300.00	N/A	25.00	995.00
Microware yearly support service (All products)				150.00
Edition Update w/o manuals 25.00		Version Update w/manuals		75.00

Special Software

STAR-DOS Level 1 (Specify ELEKTRA or DC-2) $75.00	Adaption guide		$50.00
2K MICROBUG 40.00 4K HUMBUG 75.00	Custom versions		$85.00
Spell'n Fix by Peter Stark 178.58	Write'n Spell by Peter Stark		75.11
All-In-One, Spell'n Fix, and Write'n Spell package			250.00
SUPER SLEUTH Disassembler System ($101.00 for OS-9 version)			99.00

SD/DD DISK DRIVES	1 head Tandon	2 heads Tandon	2 heads CDC	1 head MPI	2 heads MPI
30 day guarantee					
5-1/4", 40 tracks	225.00	300.00	300.00	250.00	325.00
5-1/4", 80 tracks	300.00	375.00	375.00	325.00	400.00
MPI or CDC Service Manual (Specify 40 or 80 track)		25.00	Qume DT-8		550.00

OUTBOARD EPROM PROGRAMMERS BY OPTIMAL TECHNOLOGY

Model EP-2A-79 (Personality modules extra) 169.00
Optimal Technology, Inc. 30 pin parallel I/O board for EP-2A-79 37.00
FLEX™ Software package for EP-2A-79 (Specify 6800 or 6809) 30.00
OS-9 Software package for EP-2A-79 10.00
Model EP-2B-87 (RS-232/20 MA, Motorola Imi- 8K buffer, 1200/9600 baud) 575.00
Model EP-2B-88-4 (Copies 1 to 4 EPROMS) 550.00
Personality/Copy Modules for 2708, 2716, 27C16, 2732, 27C32, 2732A, 2758, MCM68764, MCM68766, 2764, 27C64, 2764A, 27128, 27128A, 27256, 27C256, 2508, 2516, 2532, 2564, 25128, 2816, 2816A, R87C32, 8751, 38E70 $17 to $39

SPECIALS

* U.S. Robotics 1200 baud direct connect auto answer modem			299.00
* SSB BFO Floppy Disk Controller (Version 3) Run FLEX or SSB DOS			100.00
* SWTPC 4K Memory	$15.00	MP-Mb (4K bareboard)	9.95
* SWTPC MP-N (Kit)*	$54.95	S-32 RAM not included	124.50
* SWTPC MP-09 2MHz CPU*	$295.00	*While supplies last	
* High speed tape reader	50.00	300 Baud acoustic modem	129.00
* TI 810 Printer w/lower case and full vertical forms control			1200.00

SPECIAL BOARDS

Microlime II Calendar and Clock Board (Assembled) 60.00
Data Mart 16K EPROM bareboard (2708 chips) 30.00

Smoke Signal Broadcasting

DCB-4A Double Density Controller Board for 5" and 8" with DOS 549.00
SSB DOS (Specify 6800 or 6809, BFO or DCB-4A, 5" or 8") 76.00
SSB version of FLEX™ (While supply lasts) 150.00
LMB-1A Motherboard 399.00
SC8-69 6809 CPU Board 399.00
Chief 9524 64K Computer System with OS-DT-00 5" FO 4325.00
Dynamic Memory Boards M-128-X 995.00 M-256-X 1295.00 M-512-X 1895.00

HELIX

64K 6809 Computer $2395.00 Other computer systems available
DMA 5" and 8" Floppy Controller 495.00 6809 CPU Board 495.00
68008 board for SS-50 595.00 CP/M-68K 350.00

GIMIX CLEARANCE SALE	LIST	OUR PRICE		LIST	OUR PRICE
Cable (Par I/O)	24.95	20.00	6800 CPU board	224.03	100.00
Double disk reg. card	68.22	50.00	#28 controller w/Flex	328.28	270.00
64 X 16 Video Boards	198.71	100.00	Single prt ser, 1 cable	113.38	90.00
16K Mem Bds. w/chirt reg.		145.00	Dual prt par, 2 cables	138.32	110.00
93L422 DAT chip	17.50	15.00	4K PPD PROM Bd. and burner		100.00

WARNING AAA Chicago Computer Center does not provide repair or diagnostic service for customer assembled kits. AAA Chicago Computer Center does warranty and maintain service for our assembled boards. The customer should carefully take into consideration the small differential separating out kit and assembled prices when making his choice of purchase.

One other annoying gotcha. Nearly one half of this book is devoted to developing ISAM type source code files. A worthy exercise as well as a worthy addition to a C programmer's library (what this is supposed to be all about, I think). However, not all is rosy in this respect. Another library program offered in this book is called 'isamstr.h', a structure declaration file. From this book, page 206, I quote:

"The struct recv declares a structure that is used to recover data blocks which have become empty through the deletion of one or more keys. The library example in this book allows key deletion but does not list the deletion code. (This code, however, is included on the software disk that is available separately from Que Corporation.)"

The rear most part of the book list this disk, as well as some others, with convenient order cards and a 800 phone number for ordering. However, if I had been typing in all the code and using this book, as a 'library' development tool (which the title leads one to believe), and then finally arriving at page 206 only to find out that I need to invest an addition $124.95 to complete my newly developed C library function, BANG!!!

Right up front, on page one (1) they told me that the source code (all of it) is available if I don't want to type it all in, but they wait until page 206 to slip me the $124.95 shaft. NEAT!

When, and if, they get all the errors printed correctly in this book, I suspect it just might be one of your favorites. Maybe by then most can figure out how to code the missing $124.95 portion.

Oh well, after all it is tutorial, isn't it?

DMW - - -

SSB 68000

SMOKE SIGNAL
31338 Via Colinas
Westlake Village, California 91362

news release

NEW 68000 SYSTEM FROM SMOKE SIGNAL
SETS INDUSTRY PRICE STANDARD

Westlake Village, CA. With a list price of $7500, the new VAR/68K desktop computer from SMOKE SIGNA may be the lowest price fully configured 68000/UNIX computer on the market today.

A VAR/68K Model VK-5XW5 comes standard with: 256Kb RAM, one 80 track floppy, one 5Mb Winchester, eight serial and one Centronics type parallel ports, an 8 MHz MC68008 processor and Regulus, a UNIX compatible operating system produced by Alcyon. Also included in the price of any VAR/68K computer is an ergonomically-designed terminal with a green phosphor swivel screen and contoured keyboard. The system can easily be expanded to contain up to 1M bytes of RAM and sixteen RS-232 serial ports (and up to four 8 bit parallel ports). Other models in the VAR/68K series contain 1/4" streaming tape drives and Winchesters with up to 80M bytes of storage. Larger disk storage is available with standard SMOKE SIGNAL rack mount Winchesters. SMOKE SIGNAL will also offer OS-9/68000 when it becomes available and a third party will offer CP/M-68K for the VAR/68K series.

SCB-68K CPU BOARD

At the heart of this new computer is the SCB-68K CPU Board developed in-house at SMOKE SIGNAL. This standard (5 1/2" by 9") SS-50 board contains 20 address and 8 data lines and can directly address up to 1 Mbytes of RAM. The SCB-68K can plug into any 6809 based SMOKE SIGNAL computer, converting it to a 68000 based machine without changes to any current revision SMOKE SIGNAL computer boards. The SCB-68K contains a real-time clock with battery back-up, an eight-bit user programmable sense switch which can be read by software and allows custom configuring of hardware and software. The board also can contain up to 64K bytes of EPROM in two 28 pin sockets. The SCB-68K requires less than 2 amps at 5 volts.

A dynamic address translator on the SCB-68K allows for dynamic address translation on 4Kb boundaries. Each 4Kb memory segment can be write, execute and access protected. The dynamic address translator contains provision for four segment tables to co-reside in the high-speed mapping memory. This high-speed memory does not induce any wait states. Hardware circuitry provides for automatic task switching to the supervisor task table whenever the processor is executing in supervisor mode, minimizing the time required to switch between the user and supervisor modes.

The standard system contains an on-board monitor derived from MACSBUG/VERSABUG. This EPROM also contains an auto-boot loader, a single-line mnemonic assembler and disassembler and a down-loader for S1, S2 and S3 type records. SMOKE SIGNAL also offers an optional Diagnostics EPROM which contains facilities for checking RAM, I/O ports, disk controllers, the real-time clock, the dynamic address translator and the CRC's of on-board EPROM. The SCB-68K can handle external DMA transfers at up to 2M bytes per second and the processor can transfer 1.6M bytes per second in programmed I/O mode. The board contains proprietary circuitry to allow the processor to run at an even throughput during external memory accesses.

SMOKE SIGNAL also makes available an optional Test Pod/Bus-State Analyzer. The test pod plugs into a 50 pin connector on the SCB-68k allowing processor signals to be monitored on the external device. The test pod contains displays for the 20 address lines, 8 data lines, an eight-bit error latch plus various other CPU status conditions. Hardware breakpoints can be in RAM, ROM or I/O and are selected via 20 switches on the test pod. A companion single-step function can be implemented with a push button on the pod. Buttons are also provided for software Abort and hardware Reset.

REGULUS OPERATING SYSTEM

Regulus is a UNIX-like operating system which is source code compatible with UNIX V-7 and S-III and supports shared memory, record-locking, real-time tasks and contains a shell similar to the Berkeley C shell. Regulus comes standard with C, an editor, assembler, linking loader, interactive debugger and word processor and accommodates all UNIX software. Also currently available for Regulus from SMOKE SIGNAL are RM Cobol, SMC (Basic-4 type) BASIC, FORTRAN, DIBOL, Pascal, UltraCalc spread sheet, Lex word processor and Unify, the popular relational data base manager. SMOKE SIGNAL is planning on offering the Sage application generator in the near future. A version of Regulus which is source code compatible with UNIX S-V will be available soon.

Multiple user's are handled more efficiently due to the memory sharing and disk swapping capabilities of Regulus. Multiple users require only a single image of a shared program. Programs and data are automatically swapped from memory to disk allowing several users to run programs when they would otherwise exceed available memory. Disks are buffered and use internal cache in order to speed up I/O operations. There can be more than one buffer for a disk file.

Files can be protected against access by owner, public or groups. The operating system automatically prevents a user from accessing memory not owned by the user. For maximum security, password files can be read but the password itself is encrypted.

Several standard features make the operating system extremely easy to use. An on-line help facility can access disk files to retrieve portions or all of the Regulus users manual. ALIAS is a shell function which allows users to assign a name to replace a standard command name without changing disk files. HISTORY is a shell function which displays the previous 70 commands executed. The shell allows these commands to be re-executed or edited by line number, one-at-a-time or in groups. The shell supports conditionals and variables. Two processes can use named pipes to exchange data. File names can be moved from one directory to another, or within a directory. The same file can be given different names.

Regulus supports the UNIX "Mail" utilities including "NEWS". Messages written to another terminal are displayed immediately unless the user has disabled the receive message function. NEWS can be used to generate a single message which is passed to all users at time of log-on. The super-user can systematically shut-down the system, sending warning messages to all users.

Several Regulus functions allow for system monitoring. Program performance can be checked using the TIME command which returns the execution time for that program only. The disk usage command allows the user to summarize a specific directory's usage. A file system check utility allows files to be recovered; it puts the disk back in order by checking the free list against the used part of the disk plus checks for valid directory entries.

VAR/68K models which contain tape streamers can be used for image back-up and files can also be copied singly or merged into groups and then retrieved singly.

SMOKE SIGNAL will begin shipments of the first 100 VAR/68K computers in June. Delivery will be SMOKE SIGNAL's standard two to four weeks. Prices range from $7500 for the VK-5XW5 with all of the standard features described above, up to less than $20,000 for a system with 80M bytes of Winchester disk, 60M bytes of tape streamer back-up, 1M bytes of RAM and 16 ports. Call Don Simonsen at (818) 889-9340 for more information, or write SMOKE SIGNAL at 31336 Via Colinas, Westlake Village, California 91362.

- - -

HELIX **Hazelwood Computer Systems**

Computer Systems Center of Hazelwood

PRESS RELEASE

O'Fallon, Missouri
April 20, 1984
For Immediate Release

Contact: Mike Smith

Hazelwood Computer Systems is pleased to announce the availability of the MC-20 Mass Storage Controller Board which is capable of interfacing to up to 4 floppy disk drives and a SASI compatible hard disk controller.

The MC-20 is a 34 pin board which is compatible with the S-50 and S-64 busses. The extra 4 pins allow the MC-20 to address the full 16 Megabyte range of the S-64 bus.

The MC-20 is equipped with an on-board 6809 CPU running at 2MHZ with 36K RAM, up to 6K EPROM, a 6840 timer, and a 6850 ACIA (for maintenance mode)

All data as well as disk control/status information is transferred via direct memory access (DMA).

The MC-20 uses a high level protocol to communicate with the host computer which allows logical block numbers to specify the disk address. This removes the tedious and time consuming chore of logical to physical disk address calculation from the host computer. This protocol also allows the specification of multiple block data transfers.

The processing capability of the MC-20 allows other advanced features such as data buffering, hard disk segmentation, automatic motor shutdown and software settable write precompensation and motor shut-off delay.

The MC-20 will control up to 4 floppy disk drives of any size, data rate and step rate; as well as single or double density and single or double sided.

The hard disk segmentation feature allows the drive to be divided into any number of independent logical drives so that any number of operating systems may reside on a single hard disk. It is possible to load the system (boot) from any logical segment with the HELIX(TM) system.

Since the on-board 6809 controls access to both the floppy disk and the hard disk, the DMA arbitration problems found on other systems are totally eliminated thus simplifying software driver design and operation.

The MC-20 uses a high clock rate digital data separator (16MHZ) thus eliminating the need for any adjustments.

Cost is $695.00

Delivery is stock to 30 days.

Hazelwood Computer Systems is pleased to announce the availability of the Cp-68 68008 CPU Board.

The Cp-68 is a standard S-50 plug-in module which is compatible with existing S-50 and S-64 computers and associated peripheral controllers and interfaces.

The Cp-68 is jumper settable for either 4, 6, or 10 MHZ operation. There are additional jumper settings for EPROM type (2716 or 2732), battery power (on/off), and wait state insertion.

Memory management hardware consists of a base/limit register set which is active in user mode. In supervisor mode all addresses are passed through unchanged.

A battery backed-up real time clock and 2K scratchpad RAM as well as a 6840 timer are standard.

The CP-68 achieves high performance and throughput by means of a proprietary synchronization circuit which matches the asynchronous timing of the 68008 to the synchronous timing of the S-50/S-64 bus. (The CP-68 does NOT use the performance robbing VPA method of 6800 synchronization.)

Other notable features include a single cycle DMA facility (no dead cycles) and processor select logic to allow the CP-68 to reside on the bus with the CP-09 6809 board.

The CP-68 is currently available with Microware's OS9/68K Operating System.

Cost is $595.00

Delivery is stock to 30 days.

Hazelwood Computer Systems is pleased to announce the availability of the OS9/68K(TM) Operating System for the HELIX(TM) Computer System.

Hazelwood Computer Systems is proud to be the first S-50/S-64 bus manufacturer to license this fine operating system from Microware Systems Corporation of Des Moines, Iowa.

The OS9/68K system includes the operating system, a line editor, a screen editor, an assembler, and a debugger.

The OS9 operating system is the standard for 6809 systems and now is available for the 68000 system from Hazelwood Computer Systems.

OS9/68K retains all of the outstanding features of OS9 such as multi-user, multi-tasking capability and in addition allows larger user spaces and higher performance due to the processing power of the 68000 CPU.

The HELIX with OS9/68K is capable of accessing disk files from 6809 OS9 thus easing the transition to OS9/68K for 6809 users.

A 'C' compiler is available at extra cost.

Using the processor select feature of the HELIX CP-68 and CP-09 boards as well as the hard disk segmentation feature of the MC-20 controller, a HELIX system is capable of running either 6809 OS9 or OS9/68K without the need to remove the cover or change boards or disks.

Cost is $250.00

Delivery is from stock.

OS9 and OS9/68K are registered tradmarks of Microware Systems Corp.
HELIX is a tradmark of Hazelwood Computer Systems.

907 E. Terra **O'Fallon, Mo. 63366** **314-281-1055**

Dear Sir:

I saw Dan Williams's and Robert Hays's editorials in the April issue of 68MJ, which prompted me to write down my experiences with Educational Microcomputer Systems, Inc. of Irvine, California. You may have seen their small ads in Byte and other magazines.

Their M68K single board computer is a real product that works as advertised, but I would recommend it only to someone who has had a lot of experience with microprocessors and is willing to do a lot of work. What you get for your $99.00 is a pretty good printed circuit board and some of the worst documentation I've ever seen. I think the reason they call themselves Educational Microcomputer Systems is because building their system is an education. The assumption behind their documentation is that you know what you are doing.

The M68K board contains a 68000 microprocessor, two 6850 ACIAs, an 8253 programmable timer, an unlatched 16-bit bidirectional parallel port, 2K of 2114 static RAM and 8K of 2732 EPROM with a Macsbug-compatible monitor program. The RAM is addressed from $000000 to $001FFF and EPROM to $00FFFF. The PCM is addressed from $020000 to $021FFF. Because address pins A19-A23 of the 68000 aren't connected to anything, only the first 512K of the 16 megabyte potential memory map is available. Most of the first 256K of memory is dedicated to device and ROM expansion, leaving only the second 256K for RAM expansion.

Very little software is available from EMI. The Macsbug-compatible monitor is available for $135. A partial listing (actually a copy of Motorola's Macsbug documentation) is available for $15. A cross assembler that runs under CP/M is available for $149. I haven't examined the cross assembler simply because I don't have a CP/M computer. You can also get Fle-FORTH in EPROM for $110 and a FORTH assembler for $60, also in EPROM. Or you can get a version of Fle-FORTH and an assembler in listing form from Mountain View Press for $25.

I've been using my old SWTPC 6800 as a host computer for the M68K. A smart terminal program lets me upload or download programs from disk in "51-99" paper tape format, which lets me use the 6800 system like a 9600-baud teletype. I've been working my way through the Kane, Hawkins and Leventhal book, 68000 Assembly Language Programming, published by Osborne/McGraw-Hill. What I would really like to see is a 68000 cross assembler that would run under Flex 2 and not cost a fortune.

T & H Engineering, Inc.

P. O. BOX 8099 MESA, ARIZONA 85204 (602) 892-9079

PRESS RELEASE

Low Power Memory Board for EXORciser®
Features Extended Addressing and ROM Substitution.

A 64K byte static CMOS memory board, featuring low power consumption and targeted for the OEM and end user market, is being offered by T & H Engineering, Inc. The board is completely compatible with Motorola EXORciser® and Micromodule® form-factor and electrical interface. It provides extended address decoding, RAM/ROM capability, and two megahertz operation.

The M64KSE memory board comes fully populated with 64K (65,536) bytes of high speed, low power CMOS static RAM. Each of the thirty-two 2Kx8 memory devices is installed in a gold plated contact socket; any of the memory devices may be replaced with an industry standard ROM, EPROM or EPROM. The board is configured as eight seperately enabled 8K blocks for ease of integration into existing systems. Extended address decoding comes standard for use with one or two megabyte addressable systems, although the board is completely compatible with standard 64K byte systems. Fast access time allows the board to be used in one or two megahertz systems. The board requires a single 5 volt supply and typically uses only 2 watts of power. A high quality, fully solder masked board features extensive board labeling, allowing easy in-the-field board reconfiguration. Each board is covered by a one year conditional warranty.

Universal Prototype Board for EXORciser®
Features Extensive Annotation and Ground/Power Planes.

A prototype board for use in constructing user-custom circuits is being offered by T & B Engineering, Inc. The board features extensive labelling, ground and power planes, and allows use of .3, .4, .6, and .9 inch pin spaced packages.

The UE universal prototype board is completely compatible with the Motorola EXORciser* and Micromodule* form-factor and electrical interface. The board is tin-lead plated, with gold plated card edge fingers; and a fully gold plated board is also available. Heavy-duty ground and power planes are featured for low-noise. Plated-thru holes in the ground and power planes are generously distributed for installation of decoupling capacitors. All edge connector fingers are labeled with their pin number, and all device mounting holes are labeled with wire-list coordinates.

The device mounting holes are plated-thru and accept standard .025 wire-wrap pins. Forty-one vertical holes at .1 inch spacing allow two forty-pin devices to be stacked end-to-end with clearance. The board will accept devices with .3, .4, .6, and .9 inch horizontal pin spacing. Holes are provided for the mounting of card ejectors, and for chassis or stand-off mounting of the board. A continuous double-row of header mounting holes at the top of the board allows the mixed use of any sized headers. Distribution strips for plus and minus 12 volts are provided, as well as an uncommitted analog ground distribution strip.

PIXEL COMPUTER

PIXEL FEATURES

* 68000/UNIX™ Based Multi Processor Architecture
* 10 MHz Clock Speed
* .5 to 6 MB Zero-Wait-State DRAM
* Virtual Memory
* 40 - 280 MB Winchester Disk Storage
* Up to 16 Serial and 2 Parallel Ports
* Tape Cartridge Back-up
* Communications
* 13 Languages plus Translators
* Broad Selection of OEMS
* Excellent Development Tools
* Office Automation
* Word Processors and Spreadsheets
* Many Vertical Applications Packages
* Western Union and Pixel National Maintenance
* Board Swap Program
* National Advertising
* Dealer Co-op Advertising
* Sales and Service Training

For more information contact:

Laura Kildea
Marketing Communications Manager
(617) 657-8720 X2130

DEXPO Booth # 1608

FOR IMMEDIATE RELEASE

PIXEL COMPUTER OFFERS SUPERMICRO ALTERNATIVE:

UNIX PORTABILITY FOR DEC-COMPATIBLE SOFTWARE

BOSTON, MA, April 3, 1984 -- Pixel Computer will showcase its MC-68000-based, multiprocessor supermicro computers running DEC™-compatible, commercial programming languages and software application packages under UNIX™, at the DEXPO East 84 Exposition opening here today, said Paul Jolicoeur, Pixel marketing director.

Pixel will exhibit the powerful TEC Manufacturing System for online tracking and control in a variety of industrial environments, including electronics, sheet metal fabrication, and materials handling. The migration of TEC software to the Pixel 100/AP™ and Proline 80™ series computer systems represents the first implementation of a full MRP system for supermicros running under SIBOL™, a DIBOL™-compatible programming language.

Consistent with its rapidly growing third party software program, Pixel is also demonstrating vertical applications software that include programs for office productivity, general accounting, and financial modeling and analysis. Providing full portability to the multi-user 68000/UNIX environment, these "Pixel-compatible" programs offer both the OEM and end-user a low-end, state-of-the-art, price/performance alternative to software packages that previously ran only on expensive mainframes and high-end minicomputer systems.

Featured Pixel third party software and suppliers, in addition to TEC Computer Systems, include: Real World accounting software, Real World Corporation; SIBOL programming language, Software Ireland Limited; Supercomp-Twenty™ financial modeling worksheet, Access Technology, Inc.; Q-Office OAS tools, Quadratron Systems, Inc.; UNIFY™ database management system, Unify Corporation; and over thirteen languages, including FORTRAN, COBOL, PASCAL, BASIC, and SIBOL.

Pixel presents these products as an integral part of the company's commitment to the user-friendly interface shared by the 100/AP and Proline 80 series. Integrating high compute performance with interactive software and third party application packages, the 32/16-bit MC68000 microprocessor can be configured with up to 6 Mbytes of no-wait-state main memory for four to sixteen simultaneous users. The systems also provide eight RS-232 serial ports and two parallel printer ports, and optional synchronous communication subsystem. Both systems feature a menu-driven user interface that provides simple, user configurable, single-key access to system utilities and applications. The TTY-compatible ASCII terminal features an amber phosphor non-glare screen with 80-column by 24-line display, 32 programmable function keys and visual attributes for display highlighting such as underline, blink, blank, reduced intensity, and reverse video.

Consistent with the company's commitment to active product maintenance and responsive customer service, Pixel Computer's Supermicro Service Plan guarantees customers on site response within 12 hours by one of Western Union's 425 Field Service sites across the U.S. The Pixel Computer/Western Union agreement covers all parts, labor, and travel required for customer system support.

Since 1982, Pixel Computer has shipped more than 500 supermicro computer systems to locations throughout the world. Headquartered in Wilmington, MA, the company has five full service and sales offices in the United States and seven international distributorships serving Canada, Western Europe, South America and the Far East.

Dr. Larry E. Williams

Educational Microcomputer Systems can be reached at P. O. Box 14125, Irvine, Ca 92713, phone (714) 553-0113. I would suggest that you write for their current price list.

I think that the Apple Macintosh will probably become the most popular 68000 system based on price performance alone. For your $2,495 you can get an amazing amount of computer. The demonstrations given by the Computer Shoppe stores here in Nashville will knock your socks off. The rumor reported in Byte is that once 256K dynamic RAM chips become available in sufficient quantity, the Macintosh will grow from 128K RAM to 512K. I did read in the paper that Apple will sell the Macintosh to educational institutions for $1345 in sufficiently large orders. Some dealers were reported to be unhappy because that undercuts their wholesale cost. I think that Apple will get into trouble only if they completely alienate their dealers.

Your magazine is carrying a display ad on cover 2 for a new 56KPC system. Why not prevail on the folks in San Antonio to give you a write-up?

Very truly yours,

William F. Hamilton

William F. Hamilton

MOTOROLA
Semiconductor Products Inc.
P.O. BOX 20912 PHOENIX, ARIZONA 85036

AT&T VALIDATES MOTOROLA UNIX OPERATING SYSTEM

Phoenix, March 20, 1984--AT&T, originator of the UNIX® Operating System, has validated Motorola's SYSTEM V/68® Operating System, which is the first UNIX port developed jointly with an outside source.

Validation of the SYSTEM V/68 Operating System, developed in agreement with AT&T by Motorola for the M68000 family of microprocessors, means the operating system is a faithful, functional equivalent to the UNIX System V product developed for minicomputer environments.

Adaptation of the UNIX operating system by industry leaders such as Motorola, AT&T, IBM, and Digital Research is rapidly establishing it as an operating system "standard". Validation of the SYSTEM V/68 Operating System and subsequent validation of other UNIX-derived operating systems strengthens its industry-standard position.

AT&T and Motorola will market the new product under the names UNIX System V, M68000 Version Operating System and SYSTEM V/68 Operating System, respectively. The SYSTEM V/68 Operating System joins the real-time, multi-tasking VERSAdos® Operating System as another software support option on Motorola's EXORmacs® Development System and VME/10® Microcomputer System.

Support

Source code updates, maintenance, and support of the SYSTEM V/68 and UNIX System V, M68000 Version Operating systems is provided by Motorola. In addition, Motorola will provide similar support for SYSTEM V/68 object code operating on their development systems.

Price and Availability

SYSTEM V/68 Operating System is now available to manufacturers of M68000-based microcomputer systems from Motorola at both the source and object code levels which are sold individually at a cost of $2000 plus a media charge. Source code for SYSTEM V/68 is only available to holders of UNIX SYSTEM V, M68000 Version source licenses which may be obtained from AT&T Software Marketing and Sales. Object redistribution licenses for SYSTEM V/68 Operating System are available from Motorola.

For more information contact, Roger Fordham, MD-DW212, Motorola Semiconductor Products Inc., 5005 East McDowell Road, Phoenix, Arizona 85008. (602) 438-3510.

P. O. Box 61618
Sunnyvale, Ca 94088

7 April 1984

Computer Publishing Center
68 MICRO JOURNAL
5900 Cassandra Smith
PO Box 849
Hixson, TN 37343

Gentlemen:

I received my April issue and it is defective in that it has duplicate sheets with page numbers 9 through 15 inclusive and is missing sheets with Page numbers 17 through 23 inclusive. Obviously, I would like another copy.

While I am writing I might as well give you my likes and dislikes.

1. I like the honest approach you have especially in reproducing letters exactly, instead of fixing them the typical editorial white-wash and recoping.

2. I like the dedication to the 68xx world, but a few months ago a fellow wrote in and gave a very objective comparison (obviously his opinion) of Z80 and 6809. You printed it so small it was hardly readable; and cut it off in the middle. I, like like that fellow, also use a Z80 CP/M system as well as 6809 Flex and an IBM-PC. The truth is that no one designs a system, either software or hardware, to be bad; hence they each have some good features. By the same token some designs do turn out to be overall better than others. I like Flex better than either CP/M or DOS 2.0; and I like the instruction set of the 6809 (even the 6800) better than that of the Z80 or the 8088; but I won't deny that the availability of software is much greater (and truly better software exists) for the CP/M world. If I tell the truth the IBM software is gross, low in quality but high in quantity; even the most trivial program requires at least 192K bytes, an entire diskette and has a manual of at least 200 pages.

3. I dislike the space you give to the "Color Computer". It is a useless toy; inferior to the Commodore 64. The 6809 world was killed because it seems all progress stopped while all effort was directed to "getting it to go on the CoCo". Nobody with a grain of sense would every consider the Color Computer for anything serious; even if someone has got a bastardized version of Flex to work on it. Thank God the CP/M world was smarter and did not cave to a Bolt while everything was retrofit to the Timex/Sinclair.

Well, that is my soap-box.

Yours truly,

LaFarr Stuart

LaFarr Stuart

Computer Publishing Center
68 MICRO JOURNAL
5900 Cassandra Smith Rd.
P.O. Box 849
Hixson, TN 37343

[illegible] Pammel Court
Ames, Iowa 50010
April 9, 1984

Dear Sirs,

I very much enjoyed the sample issue of 68 MICRO JOURNAL which you sent me to examine. I learned more about OS-9 disks and filters in that single issue than from all I could find in the Radio Shack OS-9 manuals and all the other CoCo magazines on the market. Enclosed is my check for $24.50 for my one year subscription. Keep up the good work.

Sincerely,

Tim Harris

Tim Harris

[illegible] Corporation
[illegible] Tower 1 After Drive
Rolling Hills Estates,
California 90274
(213) 541-4828
[illegible]
Telex: 694853

NEWS RELEASE

RYAN-McFARLAND'S RM/COBOL™ AND ADVANCED RM/COS™ OPERATING SYSTEM CHOSEN FOR NCR'S NEW I-TOWER™ SMALL-BUSINESS SUPERMICROCOMPUTER SYSTEM.

Ryan-McFarland Corporation (Rolling Hills Estates, CA) is supplying their RM/COBOL™ compiler and advanced multi-user RM/COS™ operating system to NCR Corporation for use on the latter company's newly announced I-Tower™ supermicrocomputer system. NCR's 68000-based I-Tower is a business-oriented companion to the company's year-old UNIX™ based Tower™ 1632 system. NCR will sell and support RM/COS directly through their field sales organization.

This support of RM/COBOL (a GSA-certified implementation of the ANSI X3.23 74 COBOL standard) gives I-Tower users potential access to more than 750 business applications and development tools written for other systems, including NCR's own IMOS-based products. RM/COBOL runs under more than 45 operating systems on 20 different CPUs, and has been installed on more than 200,000 machines worldwide. NCR has also

been working with application developers to provide RM/COBOL and RM/COS supported packages for key vertical markets, like banking, manufacturing control, school systems, and local government.

RM/COS (Ryan-McFarland's Commercial Operating System) is multi-user and multi-tasking and is optimized for the execution of RM/COBOL programs. Because of I/O optimizations--Read and Write operations, for example, occur at the record level rather than the byte or block level--most RM/COBOL programs execute three to five times faster under RM/COS than under any other OS. Record locking, multi-user handling and contention, multiple-level access security, robust I/O (input/output) capabilities, print queue management and service, a complete JDL (job description language), terminal customization and related operations are greatly simplified by features built into the OS level with RM/COS. For all its power, RM/COS can run several users in only 256K bytes of resident memory.

Ryan-McFarland has been a leading developer of commercial systems software packages since 1970. Their software is sold either as custom-made products or standard proprietary packages to customers, including computer OEMs and sophisticated end users.

For additional information, contact Ryan-McFarland Corporation, 609 Deep Valley Drive, Rolling Hills Estates, CA 90274; (213) 541-4828.

MOTOROLA INC.
MOS Microprocessor Division
3501 ED BLUESTEIN BLVD
AUSTIN, TEXAS 78721

For further information contact

Editorial Contact: Susan Dunn
512 928-6804
Reader Contact: Jim Lovegrove
512 928-6866

MC68000 User's Manual Available From Motorola and Prentice-Hall

Austin, Texas, March 5, 1984... Motorola Microprocessor Division and Prentice-Hall Inc., publishers, announce the availability of the Fourth Edition MC68000 Microprocessor Programmer's Manual. This latest and most up-to-date design tool offers the software engineer a complete and definitive source of information on the architecture and operation of the entire M68000 microprocessor family. It includes the basic MC68000 processor, the 8-bit external bus MC68008, and the virtual memory MC68010. In addition all software in this manual will be compatible with all versions of the MC68000 family.

Each section in the Programmer's Reference Manual provides answers to the M68000's architecture, its data organization and addressing capabilities. A detailed summary of the M68000 instruction set is also included.

Each individual instruction is explained in bit pattern format and includes examples which show how each specific instruction operates and its execution time.

The manual is available for $16.95 from Prentice-Hall Inc, Englewood Cliffs, NJ, or from Motorola Literature Distribution Center, 616 W. 24th St., Tempe, Arizona, 85282.

Benchmark Report For the M68000 Family and the IAPX86 Family is Now Available

Austin, Texas, April 2, 1984..... Motorola Microprocessor Products Division announces availability of the "M68000 vs. IAPX86 Benchmark Performance." Motorola's Applications Labs have prepared the report which compares the performance differences between the segmented, dedicated register design of the IAPX family and the non-segmented, general purpose architecture of the M68000 Family.

The benchmark procedures used to compare the devices are well known and widely accepted as standard measurements of processor performance. Included are the Carnegie-Mellon/EDN benchmark set of seven procedures, the digital filter benchmark as specified by Nagle and Nelson and the Berkeley benchmark set of Patterson, Hansen, et al.

In addition to the simple comparison of execution times, the report provides a detailed analysis of each procedure, specific reasons why one device performs better than another, and a comparison of source code size and complexity for each benchmark. As a supplement to the report, the actual benchmark source code listing for the M68000 family is provided.

The report is available now and copies are available from the Literature Distribution Center, 616 West 24th St., Tempe, AZ 85287. The order number is BR150.

News Release

LPI LANGUAGE PROCESSORS INC.
400 TOTTEN POND ROAD
WALTHAM MA 02154
(617) 890-1155
TLEX 95671

FOR FURTHER INFORMATION:

Anthony Goschalk
Language Processors, Inc.
(617) 890-1155

Edgar E. Geithner
Geithner/McGowan, Inc.
(617) 875-3821

FOR IMMEDIATE RELEASE

Compilers based on Component Architecture ™
LANGUAGE PROCESSORS, INC. INTRODUCES FAMILY
OF HIGH PERFORMANCE, TRUE COMPILERS
FOR MOTOROLA MC68000-BASED UNIX SYSTEMS

WALTHAM, Mass., March 20 -- A family of high level language compilers whose Component Architecture ™ assures high performance, lets users mix sub-programs written in different languages into a single program, and whose largely identical subsystems assure reliability and quick availability was introduced today by Language Processors, Inc. The compilers, for COBOL, RPG-II, Pascal, C, PL/1, and BASIC, are aimed at computer manufacturers whose systems use the Motorola MC68000 microprocessor and the UNIX operating system.

The products are the first family of integrated compilers available from an independent software company.

They are priced at $50,000 each to hardware OEMs, plus royalties based on the size of the system on which they are used. Single copies with run-time license distribution rights are available to qualified software developers. LPI-COBOL, LPI-RPG II, LPI-Pascal, and LPI-C are available immediately. LPI-PL/I will be available this summer, and LPI-BASIC will be available in late 1984. All are available 90 days after receipt of order.

The family uses LPI's Component Architecture, a modular software architecture that combines five standard subsystems -- a front end, optimizer, code generator, run time library, and high level debugger -- into a high performance compiler. They are largely unchanged from compiler to compiler. The compilers also use the same Indexed Sequential Access Method to handle data files, so programs written in one language can address and use files written in another language.

PHILON, INC.
50 Cooper Square
New York, New York 10003
(212) 420 0317

FOR INFORMATION CONTACT:
Bob Gildenberg (212) 420-0317
Mike Morris (212) 807-0818

IMMEDIATE RELEASE

NEW HI-SPEED, PORTABLE COMPILERS FROM PHILON
FOR THE MC68000/UNIX SERIES;
HEAVY CUSTOMER SERVICE SUPPORT STRESSED

NEW YORK, March 28 -- Philon, Inc. announced today the introduction of a family of computer language compilers --PHILON FAST/Compilers -- for the MC68000/UNIX operating environments. Initial products, all of which will be available by mid-summer, are PHILON FAST/BASIC-C, PHILON FAST/BASIC-M, PHILON FAST/COBOL, and PHILON FAST/C.

"Our compilers are named 'PHILON FAST' for a reason," says Michael Parrella, president of Philon. "Benchmark results indicate superior speed of execution -- and that's what's really important today. Hardware manufacturers and software application developers recognize that in this highly competitive marketplace, fast benchmarks for their products are essential. PHILON FAST/Compilers can give them the high performance they require to help increase sales and profits."

PHILON FAST/Compilers are true compilers, not interpreters. Some of the most advanced optimization techniques have been utilized in PHILON FAST/Compilers. They were specifically designed with 16/32-bit architecture in mind and not merely moved from the 8-bit environment.

Philon's compilers are the result of 75 man-years of effort and represent a leap forward in commercial application development. They offer significant advantages in speed, portability, and productivity:

. Speed - Philon languages offer very fast execution of compiled code. Users can reduce execution time by 50% or more by simply recompiling an existing program with PHILON FAST/Compilers.

. Portability - Through Philon's development of a common intermediate code (Phi-Code), Philon's programs can now be moved easily from one hardware/operating system to another.

. Productivity - Philon languages contain a powerful set of run-time libraries and file handling routines. Also provided are numerous programming aids to minimize application development time, including an interactive debugger (the Phi-Analyser) that slashes program development time.

Philon is backing these products with a strong commitment to customer service. The company recognizes the growing demand for a support-oriented systems software organization that responds to the needs of all customers.

Says Bob Gildenberg, vice president of marketing: "Our PHILON FAST/HELP Customer Support System is fully computerized and maintains all the detailed information that must be easily accessed to assist our customers properly. Customers get prompt answers to their questions from our team of skilled customer service people."

Formed in 1980, Philon has concentrated its efforts on developing the programming software to meet the computer industry's need for a new generation of improved compilers. The company is currently under contract from a Fortune 100 corporation to develop compilers for the United States Government.

Dear Don,

This is in answer to your plea in the April issue for more info on ANY 68000 system out there. Well, I have a surprise for you. There IS a reliable, debugged, multiuser 68000 system with oodles of applications software. The name of this system is the Alpha Micro, and its operating system is called AMOS/L.

This is a relatively unknown system because: A.) the company seems to have a dislike for advertising in any microcomputer magazines; and B.) Alpha Micro computers and software are not "popularly" priced.

First, let's examine what the system is like. Alpha Micro began as a purveyor of S-100 bus computers around 1977. Its first computers were based on the WD-16 chipset, a pseudo PDP-11 chip set that had a semi-subset of the instructions available on the 68000. Converting all the old WD-16 software to the 68000 was a snap, and Alpha computers have been running an extremely large library of applications packages almost from the first day the 68000 based AM-100/L was released--in July of 1982. The top of the line AM-100/L is S-100 based. Alpha's economy model is called the AM-1000 and does not have a standard bus, although one of the many hardware vendors who support the Alpha Micro system has created a bus for it. Both computers have the potential for supporting the full 16MB address space the 68000 can handle.

But the hardware is not the best part. AMOS/L, the Alpha Micro operating system is multi-user and multitasking. It is a truly business applications oriented OS that has the power of UNIX without its incomprehensibility. It is a mature, reliable operating system with an advanced BASIC language, excellent Macro Assembler, file and record locking ability, and enough utilities to satisfy the heart of the most dedicated hacker. One of the handiest features of the system is its "virtual terminal" interface that allows applications programs to do fancy screen handling without having to worry about what kind of terminal the application is dealing with. This feature OUGHT to be a part of any multi-user system, but is sadly lacking in most, including most implementations of UNIX. Also available with AMOS/L are the Pascal, COBOL, and FORTRAN languages.

Since word processing is one of the most useful applications of any computer system, AMOS/L scores high with users in its wide range of Word Processing capabilities. Bundled with the OS is VUE, a fairly sophisticated full screen editor. Also bundled with the system is TITAN, a very sophisticated text processor capable of handling extremely complex documentation, letters, etc. Mail merge is not built into the system, but writing programs to do mail merge is simple, and many software houses have packages that can do very sophisticated mail merging. If a full word processor is what is needed, two highly sophisticated word processors, AlphaWRITE, and SuperVUE, are available. These are both complete word processing programs with "what you see is what you get" capability. They are the match of any dedicated word processor I have ever used. Many users rate SuperVUE above the WANG word processors.

As a programmer, I rate AlphaBASIC better than most COBOL implementations. It has comprehensive multi-level record definition and I/O capability. It also has a very effective ON ERROR system that catches any BASIC errors and allows the program to trap the error and take corrective action without crashing the program or mystifying the user. Another handy feature is AlphaBASIC's ability to call subprograms written in assembly language. One especially useful subroutine that I use often is called INP, and it does everything from control key trapping to input verification.

Furthermore, Alpha computers are reliable. One system that I've worked with recently has 22 terminals attached, with about 15 being used at any given time. The system runs 24 hours a day 7 days a week and the business using it is totally dependent on the computer. Even 15 minutes of downtime can be disastrous, but this system keeps going day after day after day.

I suppose the main reason most readers of your magazine haven't heard of the Alpha is because it is not a hobbyist's machine. It is a very rare individual who can afford a complete Alpha system. For businesses it is perfect, but few individual people are willing to pay for multiuser capability. Even the smallest of Alpha's computers can handle two users and a spooled printer with no add-ons or extra softwares. This is probably overkill for most hobbyists, but is just what the doctor ordered for a small business, especially one that is likely to grow. Any software written for the smallest of Alpha's computers will run totally unmodified on the largest 60 user system.

I could go on raving about extremely capable data base systems like Alpha Base that do automatic program generation, screen generation, data base security, etc. etc., but this has gone on long enough. At least I have perhaps made some people curious enough to investigate.

Sincerely,

Earl Allen
8300 Winfield Ave.
Fort Worth, TX 76109

P.S. Please renew my subscription to "68 Micro". My check is enclosed.

68' MICRO JOURNAL
5900 Cassandra Smith Rd
Hixson, Tenn 37343

Mr. Williams,

Please publish the following programs and FLEX fix. The programs should be useful to readers who like to find them have a disk full of files that need to be write or delete protected. The programs will write, delete or catalog protect the entire disk; also remove protection of the entire

disk. I don't trust those stick on write protect tabs; they do fall off with time. While writing this program I found that the FMS code 08 Put Information Record did not work; a call to TSC confirmed that it never did work. All versions of FLEX 6800 & 6809 (this is true of SWTP FLEX also) have this bug. The Fix is to patch two bytes. The Fix is: At or near $D57D find the following code BE D4 09 E6 88 22 6C 08 22 write down the address of the first byte (BE) of this code. At or near $D8B5 find the following code 8D xx xx 35 14 30 01 the xx xx is different in different versions. Now place the address you wrote down before in the xx xx part of code you just found. For SWTP FLEX 2.0:6 up to 2.9:1 the $D575 code is at $D505, and the $D8B5 code is at $D88E. If you have any questions I or Beat you call TSC. 6800 FLEX users will have to call TSC for the fix. Maybe Don Williams could call and have TSC send the info for inclusion in 68'micro. This would save TSC a lot of calls.

Thanks for a GREAT Magazine.

Joseph Aullcino

Joseph Aullcino
2014-59th Street
Brklyn, N.Y. 11204
212-837-2e87

```
                        OPT     PAG
                        WAN     PROTDISK
                ****************************************
                * Disk Protection for Entire Directory *
                ****************************************
                * By Joseph Aullcino                   *
                ****************************************
                * This program will Write, Delete or   *
                * Catalog protect the entire disk.     *
                * It will also remove protection of    *
                * entire Disk...                       *
                ****************************************

                * Syntax: PROTD<drive #>,<prot. code (W,D,C,X)>

                * Flex Equates

           CC14 BUFPNT  EQU   $CC14
           CD03 WARMS   EQU   $CD03
           CD1E PSTRNG  EQU   $CD1E
           CD27 NXTCH   EQU   $CD27
           CD42 GETHEX  EQU   $CD42
           D406 FMS     EQU   $D406
           CD3F RPTERR  EQU   $CD3F

C100                    ORG   $C100
C100 20  07     START   BRA   GETDRV
C102 01 2E 01           FCB   001,$2E,$01 Version

C105 0000       TEMP    FDB   0
C107 00         DRV     FCB   0
C108 00         PROT    FCB   0

C109 BD  CD42   GETDRV  JSR   GETHEX    get drive number
C10C 1025 00BB          LBCS  DRVERR    error?
C110 BF  C105           STX   TEMP
C113 F6  C106           LDB   TEMP+1
C116 102D 0081          LBLT  DRVERR    is drive number
C11A C1  03             CMPB  #3        between 0 and 3
>C11C 102E 007B         LBGT  DRVERR
C120 F7  C107           STB   DRV       save drive number
C123 8E  CC14           LDX   #BUFPNT
C126 BD  CD27   GETSPEC JSR   NXTCH     get protection code
C129 81  57             CMPA  #'W       is it write?
C12B 27  12             BEQ   SETW      if yes branch
C12D 81  44             CMPA  #'D       is it delete?
C12F 27  18             BEQ   SETD      if yes branch
C131 81  43             CMPA  #'C       is it catalog?
C133 27  1E             BEQ   SETC      if yes branch
C135 81  58             CMPA  #'X       is it remove prot?
C137 27  2a             BEQ   SETX      if yes branch
C139 81  0D             CMPA  #$0D      more prot codes?
C13B 27  23             BEQ   OPENDIR   if not open directory
C13D 20  E7             BRA   GETSPEC   if yes get codes
C13F F6  C108   SETW    LDB   PROT     set write prot bit
C142 CA  80             ORB   #$80
C144 F7  C108           STB   PROT


PROTDISK

C147 20  DD             BRA   GETSPCC
C149 F6  C108   SETD    LDB   PROT      set delete prot bit
C14C CA  40             ORB   #$40
C14E F7  C108           STB   PROT
C151 20  D3             BRA   GETSPEC
C153 F6  C108   SETC    LDB   PROT      set catalog prot bit
C156 CA  10             ORB   #$10
C158 F7  C108           STB   PROT
C15B 20  C9             BRA   GETSPEC
C15D 7F  C108   SETX    CLR   PROT      Clear protection
                *
C160 8E  C200   OPENDIR LDX   #FCB      point to FCB
C163 B6  C107           LDA   DRV       put drive
C166 A7  03             STA   3,X       in FCB
C168 B6  0A             LDA   #A        open directory code
C16A A7  84             STA   0,X       put code in FCB
C16C BD  B406           JSR   FMS       call FMS
C16F 26  2F             BNE   ERRA      check for error
C171 B6  07     READDIR LDA   #7        set info code
C173 A7  84             STA   0,X       put in FCB
C175 BD  D406           JSR   FMS       call FMS
C178 26  15             BNE   ERR       check for error
C17A 7D  C20C           TST   FCB+4     test for deleted
C17D 2B  F2             BMI   READDIR   directory entry (Br if yes)
C17F F6  C108           LDB   PROT      set prot code
C182 E7  0F             STB   15,X      put in FCB
C184 B6  0B     WRITDIR LDA   #B        put info record
C186 A7  84             STA   0,X       put in FCB
C188 BD  B406           JSR   FMS       call FMS
C18B 26  1D             BNE   ERRC      check for error
C18D 20  E2             BRA   READDIR   repeat t ill EOF
                *
C18F A6  01     ERR     LDA   1,X       set FCB error code
C191 81  0B             CMPA  #B        check for EOF
C193 27  10             BEQ   MSG1      if EOF branch
C195 BD  CD3F           JSR   RPTERR    if not report error
C198 7E  CD03           JMP   WARMB     return to FLEX
                *
C19B 8E  C1D3   DRVERR  LDX   #MESS1    point to string
C1 E 20  6D             BRA   OUTIT     print string

C1A0 8E  C1C6   ERRA    LDX   #MESS2    point to string
C1a3 20  68             BRA   OUTIT     print string

C1a5 8E  C1D8   MSG1    LDX   #MESS3    point to string
C1A8 20  03             BRA   OUTIT     print string

C1AA 8E  C1EF   ERRC    LDX   #MESS4    point to string
C1AD BD  CD1E   OUTIT   JSR   PSTRNG    print string
C1B0 7E  CD03           JMP   WARMS     return to FLEX

C1B3 44 72 69 76  MESS1 FCC   'Drive Number Error'
C1B7 65 20 4E 75
C1BB 60 62 65 72


PROTDISK

C1BF 20 45 72 72
C1C3 6F 72
C1C5 04                 FCB   4
C1C6 44 69 72 65  MESS2 FCC   /Directory won't open/
C1CA 63 74 6F 72
C1CE 79 20 77 6F
C1D2 6E 27 74 20
C1D6 6F 70 65 6E
C1DA 04                 FCB   4
C1DB 46 69 6E 69  MESS3 FCC   'Finished processing'
C1DF 73 68 65 64
C1E3 20 70 72 6F
C1E7 63 65 73 73
C1EB 69 6E 67
C1EE 04                 FCB   4
C1EF 43 61 6E 27  MESS4 FCC   /Can't write to Directory/
C1F3 74 20 77 72
C1F7 69 74 65 20
C1FB 74 6F 20 44
C1FF 69 72 65 63
C203 74 6F 72 79
C207 04                 FCB   4

C208                    FCB   AMB       File Control Block

                        END   START

0 ERROR(S) DETECTED


PROTDISK                                0-31-83 6:07 PM 6TSC6  PAGE    3

SYMBOL TABLE:

BUFPNT CC14  DRV    C107  DRVERR C19B  ERR    C1BF  ERRA   C1A0
ERRC   C1AA  FCB    C200  FMS    D406  GETDRV C109  GETHEX CD42
GETSPE C126  MESS1  C1D3  MESS2  C1C6  MESS3  C1D8  MESS4  C1EF
MSG1   C1A5  NXTCH  CD27  OPENDI C160  OUTIT  C1AD  PROT   C108
PSTRNG CD1E  READDI C171  RPTERR CD3F  SETC   C153  SETD   C144
SETW   C13F  SETX   C15D  START  C100  TEMP   C105  WARMB  CD03
WRITDI C184
```

HACKING THE MAC

By Michael Wolf

I am a hardware hacker. I built my first computer in 1976 from an assortment of parts I collected as evaluation samples from Motorola. I have been a Motorola chauvinist ever since. Consequently I have never before been too interested in the products of Apple Computer Inc. The first ready built computer I purchased was a Radio Shack 4K Color Computer. It was the first unit in town, and I bought it mainly on the strength of it's 6809 CPU. I took it home and didn't even plug it in before I had opened it up to see what made it tick. I upped it to 16K that afternoon and within a couple of weeks I had reverse engineered it sufficently to figure out how to up it to 64K. This I did as soon as my bank account could afford the 64K RAMS. (they were $40 each in those days).

I am telling you this to let you know what my attitude is towards computers. A computer has to be technically interesting and should have a Motorola type processor before I pay much attention to it. I look at computers from two perspectives: as a device for entertainment (can I hack it) or as a tool to do a job (can I use it to help me hack).

I was attracted to the Macintosh by an article in *BYTE* which indicated to me that it met my requirements for interest in that it had a 68000 micro and had some technically interesting features. Furthermore it might meet a need I had as a tool to do some report writing and general computering.

I went to my friendly dealer to see it and fell in love. Even if you don't anticipate purchasing one you ought to go see a demonstration. I feel that this is the advance unit that will show the way for the next generation of personel computers just as the original Apple led the way for the current generation.

The only question I had was could a hacker be happy with a computer that was obviously intended for people who didn't know anything about computers and didn't care. In short an "appliance" computer.

In keeping with this concept when I took it home I broke a long standing tradition and actually plugged the computer in before opening the case. I was so intrigued by operating it that it was several days before I opened it up.

Hardwarewise the Mac comes in three units. The main unit contains the display, computer and disc drive. The other units are the keyboard and mouse. The main unit is 11x10x13.5 inches and weighs about 16 pounds. The display is a 9 inch black and white display that is one of the sharpest I've seen. Some people have thought a 12 inch CRT would be better but I've not found it to be too small. The disc drive is a modified 3.5 inch Sony with 400K bytes capacity. It seems to be quite reliable and apparently faster than a 5 1/4 inch drive.

The detached keyboard is 13x6x2.5 inches and the mouse is 2.4x4.3x1.5 inches. The keyboard is small because it has only a standard keyboard with no numeric pad, cursor control keys, or special function keys. At first I felt this would be a serious limitation. But the mouse takes care of the need for the cursor and special function keys and a numeric pad is available as a option for those who do a lot of numeric entrys. What is left makes a nice lap sized keyboard which is easy to use and has a nice feel although it could have a lighter touch. I haven't decided if I would prefer the numeric pad be attached or not. I may make a bracket to attach them together to see how I like it.

The mouse does as much of the control of the Mac as the keyboard. In fact some applications do not even need the keyboard. Moving the cursor to select functions and clicking the single button on the mouse is all that is required to perform such things as opening files, selecting files, organizing directories, and copying files and discs. The mouse moves freely about on any reasonably clean flat surface and requires an area of about a square foot.

The interface to the operating system is through an "electronic desktop". That is a series of pictures or "icons" that represent the resources available. Icons represent application software, data files, and discs. By manipulating these icons with the mouse you can copy files, delete files, copy discs, etc. A neat feature is a selection of desk accessories such as a calculator and notepad which can be called up at any time without disturbing the current task. The icons and accessories operate out of "windows" on the screen.

While only one window is active at one time several can be present on the screen at one time. For example the windows for two discs could be on the screen at the same time. Files are copied from one disc to the other by moving the icons representating those files from one window to the other.

I was impressed by the Macintosh's human interface, but I was quite concerned that it would inhibit experenced users and become frustrating to use after some experence was gained. After several weeks and a considerable amount of work on it I am happy to report that those concerns were largely unfounded. There are shortcuts available for most functions so you don't have to resort to the mouse to do everything. However, mousing is not very time consuming and I found it simpler in most cases to use the mouse rather than memorize the various keystrokes to perform the shortcut functions. The mouse was a little awkward to use at first but as I used it more it became quite natural. The main problem is that you need quite a bit of free desk space to operate it which, to some extent, negates the advantages of the machine's small footprint. I would guess that a large aftermarket will spring up in trackballs or joysticks to replace the mouse for those people who find the mouse not to their liking.

Lest you think that I am so enamored of the Mac that have lost my objectivity I will give you some negative things that I have observed. The Mac comes with 128K of RAM and 64K of ROM. The amount of RAM is marginal. There is no clear cut way to add more except to replace the chips (4165s) with 256K RAMs. First problem is those chips cost currently $60-$90 each and you need 16 of them. Second, the RAM chips are soldered into the main circuit board. Even I have misgivings about unsoldering 16 chips from a 4 layer PCB and replacing them with $1400 worth of parts.

The single disc drive suffers from the problems any single drive system has. Copying is a hassle and it has marginally enough space for applications and data. Even though the one drive has as much capacity as the 3 drives in my CoCo system it's not adequate. Those 68000 applications programs are BIG!!

These problems are relieved somewhat by the ROM which has a lot of the utility software resident. The friendliness of the operating system also makes it less of a problem. This system is useful with only one drive, but two would be much better, and a 5 or 10MB winchester would be great.

In spite of the attention that has been paid to the software it is not completely bullet or idiot proof. It is possible to get into a corner from which the only exit is power off or reset. It is interesting to note that the reset switch is accessable only by installing a plastic clip that is provided. The manual cautions aginst installing this clip "unless you are developing applications software". Most owners won't even know what that means. Needless to say I installed the clip as soon as I found it. So far I've only crashed one disc by pushing it at the wrong time.

One thing that takes getting used to is the lack of a disc eject button. You must request the ejection of the disc via the mouse or keyboard. It is then automatically ejected after closing files etc. This is nice when copying files or when the computer wants another disc but awkward if you want to remove the disc after powering down. It contributes to the idiot proofing by preventing the removal of a disc while files are open. I guess I feel that it's taking some of my control of the system away, but I'll get used to it.

A major problem concerns the printer interface and driver software. The printer attaches to one of two serial interfaces. These interfaces are the only means of expanding the I/O capabilitys of the Mac. Not only does this limit you to only serial printers. The only printer driver software in the system is for an Apple Imagewriter. While this is a nice enough printer and not too expensive, I already owned two Epson printers. It hurt to have to buy another printer. Since the Mac does most of it's printing using the graphics capability of the printer it won't be a simple task to interface another printer.

Now a few words about software. The Mac comes with MacWrite, a word processor, and MacPaint, a graphics generator. It does not come with any form of BASIC!! A BASIC is available from Microsoft for $200.

MacWrite is a superb Word processor. The combination of mouse control and the multiple windows make it a joy to use. The only real complaints are that it only works in the insert mode, which makes it awkward to replace text, and it only works out of RAM, which limits the size of the document you can edit at one time to about 8 pages of single spaced text. It is easy to break documents into small chunks so that isn't serious. Otherwise, it is quite easy to use. It allows the changing of type fonts and sizes within a document at will. It allows justification, line spacing options, and margins. Automatic paragraph indentation, regular and decimal tabs, and page headers and footers are all provided. One of the nicest features is that any changes in typestyle or format are reflected on the screen. What you see is what you get on the printer, exactly. I wish that there were provision for using a letter quality printer but that would spoil some of the fun with typestyle/size changes.

MacPaint is a graphics generator program that has to be seen to be believed. With it you can generate pictures which can be printed out separately or incorporated in a MacWrite document. Everything except entry of text is done with the mouse. This program alone will sell a lot of Macs. I won't even try to describe it. Go see a demonstration. For entertainment value I rate it better than most computer games.

The currently available BASIC is from Microsoft and is similar to other Microsoft BASICs. This will make translation of programs from other computers relatively easy. It has a few machine specific commands such as "MOUSE" and graphic commands. There are also some syntax variations related to file handling with the Mac's file structure. The nicest feature is that it uses the Mac windows. This is especially valuable during program editing when you can have two listing windows working on separate parts of the program at once. Editing is easy as the line editor works similar to MacWrite. Lines are selected with the mouse by scrolling the listing window to display the line, selecting it with the mouse, and editing it in the command window. The command window is a single line window which is used to enter commands and programs. You can also have the listing window and the output window present at the same time which makes editing graphics easy as you can alter the program and watch the output change without a lot of messing around

with "EDIT" commands. The ease of entering and debugging programs is worth the expense of the program although I do think it should be included with the computer.

Well enough about the box as delivered from Apple. How about the hacker. Can a hacker be happy with the Mac? If you want to poke around inside the box and modify things you will be disappointed. There isn't much to do inside. No expansion slots or extra RAM sockets. About the only thing useful to be done is the RAM expansion mentioned previously. Another diffuculty is the lack of documentation. No schematics, no software entry points. The supplied manual does not even include the pinouts of the serial ports. The manual does give an address to write to to get technical documentation. I have not yet received a reply so I can't comment on what is available. The sum of this is that you might not even want to bother to open the case.

Where a hacker can really have fun is with the external ports. I already mentioned a trackball to replace the mouse. How about a printer adapter that would translate the Imagewriter commands into Epson or whatever? I would sure like to see what one of the Hewlett Packard Thinkjet printers could do on a Mac. Could one adapt a 5 1/4" drive to work off the external drive connector to facilitate downloading of software from other machines? How about the previously mentioned Winchester disc?

I expect to see some software use one of the ports as a network port to interconnect several Macs to a common printer and hard disc. Perhaps someone will come up with a I/O adapter which will communicate through the serial port and allow connection of a variety of I/O devices. There are certianly plenty of things to hack without opening the case.

I would normally take the lack of documentation as a challenge to reverse engineer the computer but, at $2500 a throw, it makes for pretty expensive mistakes. I guess I'll wait to see if Apple sends me any useful documentation before ripping into it.

In conclusion I think the Mac is the start of the next generation of personel computers. Considering that it has been on the market only a few months the amount of software and hardware support is amazing. In many ways it reminds me of the IBM PC in that it had instant support as soon as it hit the market. This indicates to me that it will be a real winner. While the software has a few bugs they are minor and a lot fewer than in any other computer I have used. The price is in the ballpark for what you get but has the potential to be cut in half and still make a fair profit for Apple. Discounts are available, particularly through universitys and to dealer's employees. If you do some careful looking you could get one for under $2K. At that price it's a real bargain.

Can a hacker be happy with the Mac? Only if they will be content with working outside the box. The development of software will be a lot of fun with the icons and windows to play with. A lot of software development tools will be available soon including at least 2 BASICs, PASCAL, and an assembler. If the ROM routines are properly documentated, and I have read that they are, software using the resources (windows, icons, mouse, etc.) will be a snap. I think the Mac will keep me entertained for some time although I will have to change my tactics somewhat. Yes, I believe the Mac can be useful and interesting for both the computer illiterate and the real hacker. Again I urge you to go see one, you'll be amazed.

Michael Wolf
3105 Arizona Ave
Los Alamos, NM 87544
April 24, 1984

6 April 1984

Dear Mr Nay:—

You ask "Where is the 68000?" and supply some remarkably good answers. Here is one more:

HSC Inc., of Herkimer, N. Y. is offering a 68000 co-processor board for Z80-based computers, with up to 768K RAM (see ads in *Microsystems*). The operating system at this writing is CP/M 68K, but that includes the Digital Research "C" Compiler which is supposed to be a full implementation fully compatible with Unix "C" so there *is* a lot of software ready for transporting.

I have seen two totally incompatible price-lists so I don't want to be too specific on that point but it looks like something between $1000 and $2000.

The point of interest for us is that they are promising to have OS-9 for the 68000 board by May. Now I'll gladly eat their whole factory without ketchup if they manage to do so, but that may mean the salvation of the 68000 is at hand?

2125 Hoogdal Road
Sedro-Woolley, WA
98284 U.S.A.

Yours very sincerely,

R. W. Odlin

R. W. Odlin

LLOYD I/O
19539 NE Gilsan
Portland, OR 97230

Frank L. Hoffman
(503) 666-1097

6809 Computer Software: Editors, Assemblers, Compilers

PRESS RELEASE

GIMIX DO !

GIMIX, Inc. has licensed "DO" from LLOYD I/O for distribution with their OS-9 systems.

DO is an OS-9 shell procedure control language. Its appearance is similar to BASIC. DO reads the procedure file from disk using a total of about 8.5K of user memory.

Parameter passing is the main feature of DO. Statements that DO does not recognize are passed to SHELL for execution. DO has 26 number variables and nine string variables. Labels are used to control the flow of execution. The ON ERROR GOTO traps errors and allows the user's procedure to take the correct action.

Don Williams, Publisher
'68' Micro Journal
P O Box 849
Hixson, Tennessee 37343

Dear Don:

For those of us who have lower case terminals, there is an error in TSC's PROT command that can be annoying, even though it is harmless. It took me a while to figure it out, since it only affects commands issued in lower case. And it isn't really annoying enough to warrant notice most of the time. The only effect is to prevent entering multiple protection statuses with a single command.

The problem is, of course, failure to force lower case commands to upper case -- but it only affects the protection codes themselves. The culprit is a branch (BCC) located at $C15E. In the version I have (Version 1), the protection code parsing loop branches to $C139 if the carry is clear.

The fix is easy. Change the BCC $C139 to BCC $C137 so the new character just fetched by NEXTCH is AND-ed with $5F, forcing upper case. The information needed to make the change:

Version 1: At $C15E, find: $24D9
Change to: $24D7

I like to keep my documentation neat, so I changed $C102 from $01 to $02 also, making the updated PROT.CMD version 2.

Now you won't have to ensure that the terminal is in upper case when you want to change a protection status to, say, WD.

Yours very truly,

Wilbur N. Killebrew, Jr.

Dear Don,
I want to report to you that John E. McGinness Jr. of 2331 Shannon Drive, Houston Texas 77027, passed away March 31,1984.

I am sure that this is not the kind of news one likes to hear about but I know that you, as well as alot of your readers, who knew John, would want to know.

John was a very kind person who, through our sometimes biweekly amateur radio net, assisted many of us through many problems. I was especially thankful since I am very isolated here in Panama.

I have been trying to find a way to let John's friends know, and I think with your assistance we can accomplish it.

Yours Very Truly,
Eric W. Davenport

Eric W. Davenport
PSC Box 3664
APO Miami, Florida 34002

Mr. Don Williams
68 MICRO JOURNAL
5900 Cassandra Smith
PO Box 849
Bixeon, TN 37343

Dear Don:

I am afraid Bob May may have his head in the sand ("Where is the 68000?", April, 1984). I know of at least forty manufacturers who are producing M68000 based computers. With rare exception, they run Unix (Trademark of Bell Labs) or a Unix clone like Idris or Regulus. Usually, they are single board or use Multi-bus or Versabus. My distributor handles two machines, the top of the line Plexus and the best value for money Altos. Both run Unix System III. I think it is interesting that Plexus first used the Z8000 and later found the M68000 the better choice, while Altos started with the 8086 (which is extremely successful for them) but found they needed the M68000 to move upscale. There are some very exciting things being done here, but they are passing you by. Yes, the chip is important, but as always, the real criteria is the software.

I first went with the M6800 because, in 1977, I could get a disk based system at a reasonable cost from Midwest Scientific Instruments. Later, I upgraded this system to run the Software Dynamics Operating System and SOLOSIC. This was darn nice software, and we used it until early 1983. During this period, we looked at the M6809, OS-9 and Uniflex. At one point we came very close to going with Gimix and OS-9. We did not because, for about the same amount of money, we could get the M68000 with a full Unix OS.

Do not under estimate Unix. People say it is not user friendly. That is bull. It is not novice friendly, but you are only a novice for a few months. Once you move out of the novice class, user friendly systems rapidly begin to get in your way. Unix is very expert friendly. Furthermore, although I have not kept up with OS-9 and Uniflex, my initial reaction was that they were a good deal less friendly than Unix. Why? Because the user interface is identical, a command interpreter called the shell, but while the Unix shell is a fairly high level interpretive language, the OS-9 and Uniflex shells are (were?) little more advanced than the SDOS command interpretor.

Unix comes with about 200 utilities, most of which can be used within a shell program. One of the most useful, at least for me, has been a program called awk, which to my knowledge only exists on Unix. Awk is an interpretive programming language with a C-like syntax which can be used to process ascii files or data. I use it like a BASIC interpretor for one-shot, quick and dirty programs. I also use it in many shell scripts where fairly extensive processing of data is required. The nice thing is, ofcos, I can do what I want, as fast as I want with available programs like awk. Later, if I find people are using a particular script often enough for efficiency to be a criteria, I can rewrite part or all of the script in C. Other useful programs are the grep family, nroff, m4, yacc, sort, sed and test. I won't describe them here. They are well covered in the literature. I will say that this letter was produced by nroff, using the ms macro package which comes with the system. The point is Unix provides the tools to do what the user wants to do, and to do it very quickly, very efficiently or in some combination of the two as desired!

Let's talk about C. Bob blames C for Lisa's problems. Not so. C is as efficient or slow as you want to make it. What is nice is, under Unix, you can profile a program to see where it is spending its time. If speed is a problem, you can then concentrate on those areas where efficiency pays the greatest return. If this means writing some routines in assembler, you can do that. The linker does not care how an object file was created.

The Unix wc program, which I imagine is similar to OS-9's count program, takes 5518 bytes, considerably less than count. On the other hand, the assembler uses 31,408 bytes, the linker 16,338 bytes, and the compiler, a huge 105,986 bytes. The Berkely editor (vi, ex, edit) also takes more than 100K. Incidentally, the three versions use the same code. C allows a program to determine by which name it was called and to act accordingly.

One must also consider the software available from other sources. This seems to be on a far higher level than the typical micro software. One example, which we have selected as the base for one of our software products is a relational database management system called Informix. Now Informix is not the very best DBMS we have seen, but it lists for less than $1000 (dbase II lists for $750). That money buys you something far more powerful than your typical micro DBMS. Schemas may be set up with the editor and changed with no difficulty. Records may be up to 2K long with keys on any number of fields. There is a default data entry language which is very nice. A C interface is provided. The report generator is provided in object form which allows you to link in custom routines. There are two query languages. One is a SQL type of language while the other is similar to QBE (Query By Example). For the latter there is a method to set up a default form, or you can build custom forms using an editor. Overall, it is difficult to find anything comparable for that amount of money.

All is not perfect. Unix needs either very fast disks or lots of RAM (preferably both) to be at its best. It is a time-sharing system and will swap programs to disk if it needs RAM. This makes spread sheet programs run slower than one might prefer. The best setup would use SMD disks. My Altos uses an Bim, Quantum winchester drive with a 55msec. average access time. The speed of this disk, compared to the typical 80-100msec. 5.25in. drive, was a critical factor in choosing the Altos. It has 0.5Mb of RAM, but the OS uses 170K, leaving only 330K for the users. By the time this is printed, it will have 1.0Mb and a second hard disk. Spreading the load across two disks also improves system speed.

It is also very important for the vendor to "tune" the system. This was Fortune's problem and it is still a bit of a problem with the Altos M68000 machines. If the system is not tuned properly, it can take inordinate amounts of time to do things. As one example, a well tuned system that takes, say, 10sec. to compile a program, may take 35sec. to do two simultaneous compiles of the programs. A poorly tuned system may take 60sec. This is not unique to Unix. I have used an H-P250 that takes 2.3 times as long to do a job if two people are using it and, of course, Bob already mentioned Lisa. Still, tuning is a poorly understood art which must be applied to get the best from the system.

Overall, I fall into the Unix as the wave of the future camp, but I mean full Unix with all the utilities. Other systems like OS-9 and Uniflex are only interesting to the extent they support a full Unix-like environment. Unix without the utilities just would not be Unix, and your readers should keep that in mind when considering Unix clones. They may have some of the features, but it is all those utilities that really make the system sing.

Sincerely,
Robert B. Peirce

Robert B. Peirce
Investment Software Co.
123 W. Edgewood Dr.
McMurrey, PA 15317

GIMIX INC. 1337 WEST 37th PLACE • CHICAGO, ILLINOIS 60609 • (312) 927-5510

GIMIX is proud to announce the availability of the

OS-9 GMX III Support ROM

On power-up or reset, the Support ROM provides OS-9 GMX III users with a command menu. The menu includes three separate memory tests that can be run anywhere in the 1 Mbyte address space to verify the integrity of the system RAM, and options to bootstrap OS-9 GMX III from either a floppy disk or directly from a hard disk.

Two versions of the Support ROM are available:

OS-9/OS-9: this version is for OS9-only systems. It includes two separate versions of the Support ROM software: the standard Support ROM as described above, and a special version that assesses the menu and tests directly from a hard disk on power-up or reset.

OS-9/GMXBUG: This version is for software switching systems that use both OS-9 and GMXBUG/FLEX. It includes the menu driven Support ROM software as described above, and a special version of the GMXBUG system monitor required to run GIMIX FLEX on GMX III systems.

Both versions are supplied on a single 8Kx8 PROM for use on the GMX 6809 CPU III. Jumpers on the CPU board determine which half of the PROM will be used on power-up or reset. Switching between OS-9 and GMXBUG/FLEX can be accomplished through jumper selection or from the keyboard. Availability is from stock.

OS-9/OS-9 Support ROM $125.00
Includes update to the Support ROM version of OS-9 GMX III

OS-9/GMXBUG FLEX Support ROM $300.00
Includes update to the Support ROM version of OS-9 GMX III, GIMIX FLEX, and VDISK (virtual disk driver for FLEX)

GIMIX is proud to announce the availability of

GIMIX O-FLEX

O-FLEX is a special version of the Technical Systems Consultants FLEX operating system, designed to run as a user process under OS-9 level II (both GMX II and GMX III versions). It allows one or more users on a level II system to run FLEX and most of the software designed to run under other versions of FLEX. O-FLEX supports all of the standard FLEX features, with the exception of print-spooling and disk formatting.

O-FLEX will read and write standard GIMIX FLEX format floppy disks and has the ability to create and use OS-9 files that are formatted and used like FLEX disks. These "pseudo-disks" are treated exactly like an ordinary FLEX disk.

Included with O-FLEX is a utility that transfers both text and binary files from OS-9 to FLEX or from FLEX to OS-9 formats, and several other utilities unique to O-FLEX. All of the applicable standard utilities normally included with standard GIMIX FLEX are also supplied.

Most commonly available software designed to run with other versions of FLEX can be run under O-FLEX, although some software may require minor modification. The patches necessary to run Technical Systems Consultants Extended BASIC (XBASIC) as well as information on patching other software for use with O-FLEX are included. NOTE: Interrupt driven software and software that directly accesses hardware other than memory (memory-management hardware, memory-mapped video, EPROM programmers, etc.) will NOT work with O-FLEX.

O-FLEX will run on both OS-9 GMX II and GMX III systems and requires 64K of free RAM for each O-FLEX user. Although primarily intended for hard disk systems, O-FLEX can also be used on floppy-only systems. Note: While O-FLEX can read and write FLEX format floppy disks, a stand-alone FLEX system is required in order to format disks.

GIMIX O-FLEX $250.00
Includes O-FLEX on OS-9 format disk, and manual

GIMIX will again be exhibiting at NCC on July 9 - 12, in Las Vegas. Our booth numbers are 34016/18. We look forward to seeing 6809 users and showing them our latest state-of-the-art products.

After convention hours, we will have our informative, fun, creative 6809 Mini convention, in the GIMIX Hospitality Suite at the Las Vegas Hilton.

Dear Sir;
I constantly read in computer magazines that "X number of sales have been lost because of copying." Has anyone ever considered how many sales have been made because of copying. A lot of programs are just to expensive for many hobbyists to afford unless they go in with a few friends- its either copy or do without. And the problem is worse on the S-50 bus than for people with IBM'S or Apples; they have a large number of public domain and "el cheapo" programs to work with. We start out paying four or five hundred for one language and an operating system and then another good thousand for a few basic programs. That would be fine for a business where the programs are used every day- but $250 for a program to use two or three hours a month?
I'm not advocating stealing- in fact I' spent a lot of time arguing for honesty with people who are copying music (where the problem is even worse). But both the software people and the music publishing houses are going to have to make some provision for the small operator who needs a good product at reasonable cost.
I don't know what the solution is but I am quite certain a lot of copies aren't lost sales, they're programs that never would have been used otherwise. Thanks for the OS-9 articles in 68 Micro. The system is starting to make sense.
Robert J Otto
Box 60
Crystal ND 58222

HELP

Sirs;
I am interested in putting together a high-resolution color-graphics system based on a 64K or 128K home type computer for scientific purposes. Any ideas or information about your products you could send me would be appreciated. I would like to interface a video camera with the computer also. What will I need to do this?
Thanks for any help.
Marathon Petroleum Egypt
539 S Main St
Findlay OH 45840
Norman Mark

6809 COMPLETE SYSTEM 6809

ADVANCED TECHNOLOGY ECONOMICAL

The PT69 COMPLETE SYSTEM features the proven PT69 single board computer. Powerful performance + reliability - an unbeatable combination! Complete Systems feature:

- Double Sided/Double Density Drives (40 or 80 track) Cabinet, Power Supply
- Optional CRT and Printer
- 1MHz 6809E Processor
- 2 RS232 Serial Ports
- 2 8-Bit Parallel Ports
- 56K RAM: 4K EPROM: 4K I/O

* COMPLETE SYSTEM with PT69 Board, 2 DS/DD 5¼" 40 Track Drives, Cabinet, and Power Supply. $999.95
* PT69, Assembled, Tested, with Power Supply and Cabinet. $399.95
* PT69, Assembled and Tested. $299.95

— Kits are available —

— Write for information on CRT and Printer —

OEM Inquiries Welcome

PERIPHERAL TECHNOLOGY
"Supplying Your Computer Needs Since 1978"
3760 Lower Roswell Road
Marietta, Georgia 30067

VISA/MASTERCARD/CHECK/COD 404/973-0042

CLASSIFIED ADVERTISING

TELETYPE Model 43 PRINTER - with serial (RS232) interface, and full ASCII keyboard. **LIKE NEW** - New cost $1295.00 - ONLY **$759.00** ready to run - Call Tom - Larry - Bob, CPI 615 842-4600

***

SWTP 6800- GIMIX 32K static, MPA CPU, Peripheral Tech FD-1A Disk Controller, MPC Serial, MP-L Parallel, MP-N Calculator, CT-64 Terminal, AC-30 2 Cassette I/F. Flex, Editor, Assembler, Utilities, Text Processor, Pilot, Basic. All Documentation, Hardware, Software-$800 or best offer.
813-839-7225 Harry after 6:30 PM EST.

***

For Sale: SWTPC 6809 System. Uniflex OS. Includes: 128K Ram; 20 Megabyte Hard Disk; two 8212 terminals, Dual 8" DSDD Drives; Miscellaneous Software, Furniture (Desk, Stand). Less than 1000 hours on time. 1 1/2 years old. Excellent condition. Will consider selling individual pieces. Make me an offer.
Call Marilyn (512)250-9844, Business Data Systems, 8300 Pilgrims Pl., Austin TX 78759

***

STWPC 6809, 56K,serial, parallel, clock, DSDD controller, 2 Qume DT-8, H-19, Centronics 306 w/lower case, truck load of software (over $1600)- a steal at $2500, make me an offer. Complete set of Micro Journal from V1.1 to present,offer.
(503)647-5878 Don Kinzer

***

Flex System-Elektra Cabinet, two Perteck DS/DD Drives, SWTP MP-09 cpu and DC-4 controller, Digital Research 64K Ram/Rom card, Thomas 300 baud modem card, Alford sp ch synthesizer. Software and documentation included. $1500.00.
Bill Medlinski (201)772-5975

***

For Sale: Complete 6809 system. SWTPC mainframe, Percom 6809-CPU, 80 track disks, FLEX09, XBASIC, Text processor. Many othe goodies.
S Brown, 35 Kettle Pond Rd, Amherst, MA 01002

***

Original SWTPC cassette 8K Basic, Co-Res Assembler & Percom Cis-30+ cassette/terminal interface, full documentation. Various other software on cassette, MSI Basic, Space Voyage etc. Make me an offer.
(505)298-9187 Richard Carreras

***

Two complete SWTP-6809 Flex/Uniflex systems:
System-1: S/09+, 256K, 10meg Mini Winl, X-12, 8212. 35% off retail. $7,900.
System-2: S/09, 192K dual 8-inch diskettes, ct-82, 8212, Qume 45cps printer with tractor, sheetfeeder, and MP-QP interface board. 45% of retail. $7,500.
Make offers to Richard Davidson, (517)332-5989.

***

New Verbatim Disks. Never been opened. I bought these from an office supply house that went out of business.

5"

4 Boxes 5" 525-10 SSDD 10 Sector $19.00 Box
4 Boxes 5" 525-16 SSDD 16 Sector $19.00 Box
30 Boxes 5" 550-01 DSDD Soft Sector $24.00 Box

8"

27 Boxes 8" 34-4001 DSDD Soft Sector $27.00 Box
29 Boxes 8" 34-8000 SSDD Soft Sector $24.00 Box
3 Boxes 8" 34-9000 SSSD Soft Sector $23.50 Box
3 Boxes 8" 65-1000 SSSD 32 Sector $29.00 Box
Tom Williams 1-800-338-6800

***

68 MICRO JOURNAL PROGRAMS - DISK

Disk-1 Filesort, Minicat, Minicopy, Minifms, **Lifetime, **Poetry, **Foodlist, **Diet.
Disk-2 Diskedit w/ inst.& fixes, Prime, *Prmod, **Snoopy, **Football, **Hexpawn,**Lifetime
Disk-3 Cbug09, Sec1, Sec2, Find, Table2, Intext, Disk-Exp, *Disksave.
Disk-4 Mailing Program, *Finddat, *Change, *Testdisk.
Disk-5 *DISKFIX 1, *DISKFIX 2, **LETTER, **LOVESIGN, **BLACKJAK, **BOWLING.
Disk-6 **Purchase Order, Index (Disk file indx)
Disk-7 Linking Loader, Rload, Harkness
Disk-8 Crtest, Lanpher (May 82)
Disk-9 Datecopy, Diskfix9 (Aug 82)
Disk-10 Home Accounting (July 82)
Disk-11 Dissembler (June 84)
Disk-12 Modem68 (May 84)
DISK-13 *Initmf68, Testmf68, *Cleanup, *Dskalign, *Leobug, Help
Disk-14 *Init, *Test, *Terminal, *Find, *Diskedit, Help

NOTE:
This is a reader service ONLY! No Warranty is offered or implied. The Disk Files are as received by **'68' Micro Journal**, and are for reader convenience ONLY (some MAY include fixes or patches). Also 6800 and 6809 programs are mixed, as each is fairly simple (mostly) to convert to the other.

PRICE: 8" Disk $29.95 - 5" Disk $24.95

68 MICRO JOURNAL
POB 794
Hixson, TN 37343
615-842-4600

* indicates 6800; ** indicates BASIC SWTPC or TSC
6809 has no indicator.

MASTER CARD - VISA accepted
Foreign -- add **10%** for surface
or **20%** for air!!

COMPILER EVALUATION SERVICES
By: Ron Anderson

The S.E. MEDIA Division of Computer Publishing Inc.
Is offering the following 'SUBSCRIBER SERVICE':

COMPILER COMPARISION AND EVALUATION REPORT

Due to the constant and rapid updating and enhanchment of numerous compilers, and the different utility, appeal, speed, level of communication, memory usage, etc., of different compilers, the following services are now being offered with periodic updates.

This service, with updates, will allow you who are wary or confused by the various claims of compiler vendors, an opportunity to review comparisons, comments, benchmarks, etc., concerning the many different compilers on the market, for the 6809 microcomputer. Thus the savings could far offset the small cost of this service.

Many have purchased compilers and then discovered that the particular compiler purchased either is not the most efficient for their purposes or does not contain features necessary for their application. Thus the added expense of purchasing additional compiler(s) or not being able to fully utilize the advantages of high level language compilers becomes too expensive.

The following COMPILERS are reviewed initially, more will be reviewed, compared and benchmarked as they become available to the author:

PASCAL "C" GSPL WHIMISCAL PL/9

Initial Subscription - $39.95
(Includes 1 year updates)
Updates for 1 year - $14.50

S.E. MEDIA - CPI
5900 Cassandra Smith, POB 794
Hixson, TN 37343
615 842-4601

GOOD NEWS!

C for the 6809 WAS NEVER BETTER!

INTROL-C/6809,Version 1.5

Introl's highly acclaimed 6809 C compilers and cross-compilers are now more powerful than ever!

We've incorporated a totally new 6809 Relocating Assembler, Linker and Loader. Initializer support has been added, leaving only bitfield-type structure members and doubles lacking from a 100% full K&R implementation. The Runtime Library has been expanded and the Library Manager is even more versatile and convenient to use. Best of all, compiled code is just as compact and fast-executing as ever - and even a bit more so! A compatible macro assembler, as well as source for the full Runtime Library, are available as extra-cost options.

Resident compilers are available under **Uniflex, Flex** and **OS9.**

Cross-compilers are available for **PDP-11/UNIX** and **IBM PC/PC DOS** hosts.

Trademarks:
Introl-C, Introl Corporation
Flex and Uniflex, Technical Systems Consultants
OS9, Microware Systems
PDP-11, Digital Equipment Corp.
UNIX, Bell Laboratories
IBM PC, International Business Machines

For further information, please call or write.

INTROL
CORPORATION
617 W. Virginia St.
Milwaukee, WI 53204
(414) 276-2937

HELIX...OS9/68K...NOW!

HAZELWOOD COMPUTER SYSTEMS is pleased to announce the availability of several new products: the CP-08 68008 CPU board, the MC-20 Disk Controller, and OS9/68K(TM) for the HELIX(TM) computer system.

The CP-08 68008 CPU board is a standard S-50 plug-in capable of 4,8,or 10MHZ operation. Featues include a 6840 Timer, OKI battery back-up real-time clock, Base and Limit Register, and the capability to co-reside on the same bus with the Hazelwood CP-09 6809 board. The CP-08 achieves high performance by use of a proprietary synchronization circuit to match the asynchronous timing of the 68008 to the S-50 bus.

The MC-20 Disk Controller Board is capable of controlling up to 4 floppy disk drives (either 5 1/4 or 8 inch in any combination) as well as any SASI compatible hard disk controller such as the XEBEC 1410. The MC-20 has its own 2MHZ 6809 CPU on board with 56K RAM, up to 6K EPROM, a 6840 Timer, and a 6850 ACIA for maintenance mode. All data as well as control/status transfers are via direct memory access (DMA).

The OS9/68K operating system is the 68000 version of the popular operating system by Microware Systems Corporation. Included with the operating system is a line editor, a screen editor, an assembler, and a debugger. A 'C' compiler is available at extra cost. Current 6809 OS9 users will appreciate the features of this system running on the HELIX including higher speed and larger work area as well as the ability to access 6809 OS9 disk files.

As a one-time, introductory offer, Hazelwood Computer Systems is offering the following package discounts to enable more users to move up to the computing power of OS9/68K. This sale ends June 30, 1984.

Package 1

CP-08 68008 CPU Board
MC-20 Disk Controller (695.00)
OS9/68K Operating Sys (250.00)

Total Retail Value $1540.00
Sale Price $995.00

Package 2

CP-08 68008 CPU Board (595.00)
MC-20 Disk Controller (695.00)
DM-256 256K Memory (995.00)
OS9/68K Operating Sys (250.00)
Total Retail Value $2535.00
Sale Price $1795.00

Package 3

HX-08256 256K 68008 Helix (2995.00)
MC-20 Disk Controller (695.00)
OS9/68K Operating Sys (250.00)

Total Retail Value $3940.00
Sale Price $3195.00

Package 4

CP-08 68008 CPU Board (595.00)
OS9/68K Operating Sys (250.00)

Total Retail Value $845.00
Sale Price $595.00

Order from:

HAZELWOOD COMPUTER SYSTEMS
907 E. Terra Lane
O'Fallon, MO 63366
(314) 281-1055
Or the authorized HELIX Dealer in your area.

OS9 and OS9/68K are registered trademarks of Microware Systems Corp.
HELIX is a trademark of Hazelwood Computer Systems.

TOLL FREE
1-800-338-6800
for Ordering

SOUTH EAST MEDIA

5900 Cassandra Smith Rd.
Hixson, TN 37343
for information
call (615) 842-4601

CoCo OS-9™ FLEX™
SOFTWARE

ASSEMBLERS

Southeast Media

ASTRUK09

A "Structured Assembler for the 6809" which requires the TSC Macro Assembler. Allows direct use of structured statements such as IF, ELSE, DO, REPEAT, etc., and provides indented level formatting of the listing so that the structure is apparent. Re. '68' Micro Journal, Sept. '83 (program was called "STASM09"; has been renamed due to conflicts).

A User reports

"... I'm very pleased and am now writing almost exclusively in (ASTRUK09). I've selected it over --- for all future systems development... As (one) of my early evaluations, I rewrote a rather elaborate routine originally done in assembly. Out of the 1000 bytes of code generated, the (ASTRUK09) version used only 20 more bytes than the original. --- could not andle this program since it uses triple-precision fixed point arithmetic... I have a large body of code already written t at is incompatible with --- constructs. No problem with (ASTRUK09) and the structure sure helps in understanding the logic!"

F, CCF - $99.95

TSC

Macro Assembler

The FLEX STANDARD Assembler. F,CCF $50.00

Relocating Assembler w/Linking Loader

Use with many of the C and Pascal Compilers. F,CCF $150.00

Great Plains Comp. Co.

RELMAC

Relocating, Recursive-Macro Assembler and Linking Loader.
F,CCF $120.00; w/Source $240.00

OmegaSoft

- MRALLI

Relocating Assembler and Linking Loader
F,CCF $125.00; for One Year Maint., add $50.00

Windrush Micro Systems

MACE, by Graham Trott. F,CCF - $98.00

Computer Systems Consultants

SUPER SLEUTH

Computer Systems Consultants Super Sleuth is a "Time Tested", reliable, PROVEN Disassembler that has gained acceptance through out the SS-50 Bus Community as an extremely POWERFUL, INTERACTIVE, Software Tool. The Super Sleuth Software Package consists of 3 Programs; SLEUTH (the Disassembler), CHGNAM (used to globally Change Labels to a meaningful Name), and XREF (a Cross Reference Generator for Source Code Files). SLEUTH will Disassemble Memory Resident 6809 Code and 6800, 6801, 6802, 6803 (the "Baby CoCo"), 6805, 6808, 6809, and 6502 (Apple, Atari, Commodore, etc.) Binary Disk Files. (See Aug. '83 '68' Micro Journal "Color Users Notes" Column for a full Review.)

Color Computer	SS-50 Bus (all w/ Source)
CCO (32K Req'd)	
Obj. Only $49.00	F. $99.00
CCF, Obj. Only $50.00	U. $100.00
CCF, w/Source $99.00	O, $101.00
CCO, Obj. Only $50.00	

ALL Computer Systems Consultants Software
runs on the Color FLEX Systems
ALL in stock
call 800-338-6800
for SAME DAY DELIVERY

Computer Systems Center

DISMANTE +

An "easy to use", powerful Disassembler for Disk Resident 6809 and 6800 Binary Files. Allows the development of a "Control File" of various Program "Boundaries" during successive disassemblies; can use a Label File which automatically replaces a Hex location with a Label Name; Includes an XREF Utility; etc. Label Files provided for Mini-FLEX, FLEX2, FLEX9, Color Computer (for use with Color FLEX Systems), etc. OS-9 Version includes special OS-9 options.

CCF, Obj. Only $100.00
CCO, " " $150.00
F, " " $100.00
O, " " $150.00
U, " " $300.00

COMPILERS & DECOMPILERS

6809 "Structured" Assembly Lang. Compilers

Windrush Micro Systems

FL/9

By Graham Trott. A "Structured" Assembly Language Editor/Compiler/Debugger, all in ONE PACKAGE: provides a totally INTERACTIVE Program Development Cycle. The Compiler supports large Symbol Names, Variable Types, Pointers, Control Structures, Stack, A-,B-, and D-Register manipulation, etc. The Source-Oriented Trace/Debugger provides Single Stepping, Breakpointing, etc. An excellent Software Development Tool for utilizing the power of the 6809 in developing small to medium sized packages.

F, CCF - $198.00

Whimsical Developments

WHIMSICAL

Need the Ease of Design and Maintainability of "Structured Programming" AND the Speed and Control of Assembly Language? Then WHIMSICAL was designed for you! This Single Pass, Recursive Descent Compiler provides the tool for developing simple Utilities to MAJOR Systems in Assembly Language. Supports 3 "Lex" Levels which allow one level of Procedure nesting, or more within "Modules". It is easy to develop programs written for other machines since you are working at the Assembly Language level. Features unified, user-defined I/O; produces ROMable, relocatable, recursive, re-entrant Code; Structured style and statements with Procedures and Modules; supports Byte and Double-Byte primitives with 3 types of Integers (up to 32 bit), Char and Boolean, and unlimited mixed Arrays (vectors only); Interrupt handling; unlimited length Variable Names; Variable Initialization (defaults to $00); Include "Source File" directive; Conditional compiling; direct Code insertion; control of the Stack Pointer; etc. To quote Ron Anderson in his review of WHIMSICAL in the Sept. '83 Issue of '68' Micro Journal that, except for the lack of floats, "...., I have to give this one VERY high rating. ...". It is a FAST Compiler which produces FAST Code (his "Primes" Benchmark ran at 9 secs. on a 2 MHz System).

F and CCF - $195.00

'C' Compilers

Windrush Micro Systems

C Compiler

By James McCosh. Full featured C Compiler for the FLEX Operating System. Includes a Reloc. Asmb., but needs the TSC Relocating Assembler/Linking Loader (which includes a Library Manager) for those "full blown" System Packages.

F and CCF - $295.00

Introl

C Compiler

A full-featured C, streamlined for the 6809. Generates very efficient object code. Output "benchmarks" close to 10MHz 68000 in 8 Bit Operations; 1.5 times faster than a 4 MHz Z80 when using a 2MHz 6809 System (Re. p 43, "68" Micro Journal, May '83). Floats, etc.

F, CCF, and O - $375.00
U - $425.00
One Year Maint. - $100.00

VISA MasterCard

"FLEX is a trademark of Technical Systems Consultants
"OS9 is a trademark of Microware

TOLL FREE
1-800-338-6800
for Ordering
SOUTH EAST MEDIA
5900 Cassandra Smith Rd.
Hixson, TN 37343
into (615) 842-4601
CoCo OS-9™ FLEX™
SOFTWARE

Availability Legends —
F = FLEX, CCF = Color Computer FLEX
O = OS-9, CCO = Color Computer OS-9
U = UniFLEX
CCD = Color Computer Disk
CCT = Color Computer Tape

PASCAL Compilers

TSC

PASCAL Compiler

Native Code Compiler (UCSD Oriented).

F and CCF - $200.00

Lucidata

PASCAL Compiler

P-Code Compiler (ISO Standard). Designed especially for Microcomputer Systems: Run-time System checks available resources for each task, allowing operation on even minimal computer systems. Allows linkage to Assembler Code for maximum flexibility.

F and CCF 5" - $190.00
F 8" - $205.00

OmegaSoft

PASCAL Compiler

For the PROFESSIONAL; ISO Based, Native Code Compiler. Primarily for Real-Time and Process Control applications. Use custom I/O devices in place of the Pascal INPUT and OUTPUT; Long Int. (32 Bit); Dynamic length strings; Interrupt processing, ROM-able, PIC, Re-Entrant Code, etc. POWERFUL! Includes Source for the Symbolic Debugger, Runtime, and several Utilities. Requires a "Motorola Compatible" Relocating Assembler and Linking Loader.

F and CCF - $425.00
One Year Maint. - $100.00

DECOMPILERS

Southeast Media

DUB (A UniFLEX "basic" De-Compiler)

Re-Create a Source Listing from UniFLEX Compiled **basic** Programs. Easy to Use; works w/ **ALL** Versions of UniFLEX basic; Output to Disk or Terminal. Time TESTED and PROVEN; SOLID!

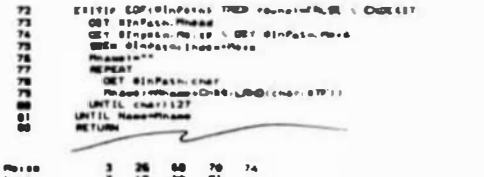
U - $219.95

UTILITIES

Southeast Media

Basic09 XRef

This Basic09 Cross Reference Utility is a Basic09 Program which will produce a "pretty printed" listing with each line numbered, followed by a complete cross referenced listing of all variables, external procedures, and line numbers called. Also included is a Program List Utility which outputs the listing without the overhead of building the cross reference table, which allows it to run considerably faster when only a "pretty printed" listing with line numbers is desired. Requires Basic09 or Run8 for operation.



O and CCO - Obj. Only -- $39.95
O and CCO - w/ Source - $79.95

Southeast Media

OS-9 VDisk

Give your OS-9 Level I System the speed of memory access that can be several orders of magnitude over your present floppy disk drive. Use that Extended Memory capability of your SWTPC or Gimix CPU card (or any other that has the same format DAT). The size of the Virtual Disk is completely variable in whole increments of 4K up to 960K, which is all that these systems can address beyond the base page that OS-9 Level I uses. By putting all of your CMDS Directory on your Virtual Disk, you can have the fastest execution speed possible (next to eating up System Memory with all of them). You can also set up high speed inter-process communications via random virtual disk files and not eat up valuable system memory with pipe buffers. Some Assembly Required - **Level I ONLY.**

O, obj. only - $79.95
w/ Source - $149.95

TOLL FREE
1-800-338-6800
For Ordering

SOUTH EAST MEDIA

5900 Cassandra Smith Rd.
Hixson, TN 37343
for information
call (615) 842-4801

CoCo OS-9™ FLEX™
SOFTWARE

Southeast Media

O-F

---- OS/9 to FLEX - FLEX to OS/9 ----

Finally; the barrier has been removed between OS/9 and FLEX formatted disks! Now you can READ **from**, and WRITE **to**, a Single Sided 5" or 8" FLEX diskette from OS-9 with O-F. O-F is a new and unique program, written in BASIC09 (with Source), that performs the following functions;

REFORMAT: A BASIC09 Program that reformats a chosen amount of an OS-9 disk to FLEX Format so it can be used normally by FLEX.

FLEX: A BASIC09 Program that does the actual read or write function to the special O-F Transfer Disk, all selected from a user-friendly menu. Functions provided include reading the FLEX Directory, Deleting FLEX Files, Copying both directions, etc. All selections are interactive and complete, including all necessary prompts to the operator.

FLEX users can read, write and use the special disk as any other FLEX disk, provided the FLEX directory is not allowed to continue beyond track zero (too many files).

F and CCF - $79.95

Southeast Media

COPYMULT

--- Copy **LARGE** Disks to several **smaller** disks ---

The following FLEX utilities allow the backup of **ANY size disk** to any SMALLER size diskettes (Winchester to 8's or 5's, 8" to 5's, etc.). By simply inserting diskettes as requested by COPYMULT, a large disk system may be downloaded to your present floppy disk system, any size. No need to fiddle with directory deletions or any of the other tedious operations that must be done using the normal copy routines.

COPYMULT.CMD understands normal "copy" syntax and always keeps up with files already copied by maintaining directories for both host and receiving disk system, eliminating hours of tedious keyboard entries and other time consuming cleanup chores.

BACKUP.CMD is a special program that downloads "random" type files, any size.

RESTORE.CMD a special program to restructure copied "random" files for copying, or recopying back to the host system.

FREELINK.CMD a "bonus" utility that "relinks" the free chain of floppy or hard disk thereby eliminating fragmentation.

Completely documented source files included.

ALL 4 Programs (8" or 5") **$99.50**

Southeast Media

CHESS 6809

Requires FLEX and DISPLAYS On Any Type Terminal
Features:

*Four levels of play.
*Swap side. *Point scoring system.
*Two display boards. *Change skill level.
*Solve Checkmate problems in 1-2-3-4 moves.
*Make move and swap sides. *Play white or black.

This is one of the strongest CHESS programs running on any microcomputer, estimated USCF Rating 1600+ (better than most 'club' players at higher levels).

F and CCF - $79.95

VISA MasterCard

*FLEX is a trademark of Technical Systems Consultants
"OS9 is a trademark of Microware

TOLL FREE 1-800-338-6800 For Ordering
SOUTH EAST MEDIA
5900 Cassandra Smith Rd. Hixson, TN 37343 Info (615) 842-4801
CoCo OS-9™ FLEX™ SOFTWARE

Availability Legends —
F = FLEX, CCF = Color Computer FLEX
O = OS-9, CCO = Color Computer OS-9
U = UniFLEX
CCD = Color Computer Disk
CCT = Color Computer Tape

TOLL FREE
1-800-338-6800
For Ordering

SOUTH EAST MEDIA

5900 Cassandra Smith Rd.
Hixson, TN 37343

for information
call (615) 842-4601

CoCo OS-9™ FLEX™
SOFTWARE

Southeast Media

DIET-TRAC Forecaster

DIET-TRAC Forecaster is an XBASIC program that plans a diet in terms of either calories and percentage of carbohydrates, proteins and fats (C P G%) or grams of Carbohydrate, Protein and Fat food exchanges of each of the six basic food groups (vegetable, bread, meat, skim milk, fruit and fat) for a specific individual.

Sex, Age, Height, Present Weight, Frame Size, Activity Level and Basal Metabolic Rate for no mal individual are taken into account. Ideal weight and sustaining calories for any weight of the above individual are calculated. When a weight goal is given (either gain or loss), and a calorie plan is agreed upon between the computer and the individual, the number of days to reach the weight goal is projected. The starting and ending rate of weight loss is calculated, and a daily calendar with each day's weight for a 30-day period is printed.

F - $59.95
U - $89.95

Southeast Media

XDATA
A COMMUNICATION Package
for the UniFLEX Operating System

Allows UniFLEX Based Systems to Transmit and Receive files to and from other Computer Systems via Modem. Use with CP/M, Main Frames, other UniFLEX Systems, etc.

-- Verifies Transmission Integrity using checksum or CRC
-- Automatically Re-Transmits bad blocks
-- Transmits data in 128 byte blocks

U - $299.99

Southeast Media

JUST
Text Formatter

JUST, a Text Formatter developed by Ron Anderson, provides numerous features which make it a valuable addition to any FLEX Users Software Library. JUST is designed for formatting Text Output for Dot Matrix Printers and provides many unique features:

-Output the "Formatted" Text to the Display for format analysis and change.
-Output the "Formatted" Text to a Text File for use with the supplied FPRINT.CMD for producing multiple copies of the Text on the Printer INCLUDING IMBEDDED PRINTER COMMANDS (this Utility is very useful at other times also, and worth the price of the program by itself).
-"User Configurable" for adapting to other Printers (comes set up for Epson MX-80 with Graftrax); provides for up to ten (10) imbedded "Printer Control Commands", such as Italics on and off, boldface on and off, etc.
-Automatic compensation for a "Double Width" printed line.
-Includes the normal line width, margin, indent, paragraph, space, vertical skip lines, page length, page numbering, centering, fill, justification, etc.
-Use with ANY Editor.
-Supplied with "Structured Source" (Windrush PL/9); easy to see the flow of the program.

F and CCF - $49.95

Lucidata

PASCAL UTILITIES
Requires LUCIDATA Pascal ver 3.

XREF -- produce a Cross Reference Listing of any text; oriented to Pascal Source.

F and CCF - $25.00

INCLUDE -- allows the inclusion of other Files in a Source Text; has unlimited nesting capabilities. Also allows Binary File inclusions.

F and CCF - $25.00

PROFILER -- produces an Indented, Numbered, "Structogram" of a Pascal Source Text File. Allows viewing the overall structure of large programs, and provides clues as to the integrity of the program. Supplied as Source Code; requires compilation.

F and CCF - $25.00

Lucidata

COPYCAT
Pascal NOT required

Allows reading TSC Mini-FLEX, SSB DOS68, and Digital Research CP/M Disks while operating under FLEX 1.0, FLEX 2.0, or FLEX 9.0 with 6800 or 6809 Systems. COPYCAT will not perform Miracles, but, between the program and the manual, you stand a good chance of accomplishing a transfer. Includes Utilities to List Directories, Copy Files, and convert Text Files when required. Also includes a Utility for investigating Physical Compatibility problems. Programs supplied in Modular Source Code (Assembly Language) to make it easier to solve unusual problems.

F and CCF 5" - $50.00
F 8" - $65.00

Computer Systems Consultants

FLEX DISK UTILITIES

Eight (8) different FLEX Utilities that should be a part of every FLEX Users Toolbox; Assembly Language (Source Code):

Copy a File with CRC Errors, so it can possibly be salvaged; Test Disk for errors; Compare two Disks; a fast Disk Backup Program; Edit Disk Sectors; Linearize Free-Chain on the Disk; print Disk Identification; and Sort and Replace the Disk Directory (in sorted order).

F and CCF - $50.00

WORD PROCESSORS

Alford and Associates

SCREDITOR III

EXTREMELY Powerful Screen-Oriented Editor/Word Processor. Almost 50 different commands; EXCELLENT Documentation (over 300 pages), including a full Tutorial Section to help you learn how to use the system. Features Cursor-based editing, dynamic Screen Formatting (what you see is what you get), Multi-Column display and editing, "decimal align" columns (AND add them up automatically, if wanted), define multiple keystroke macros, even and odd page number headers and footers, imbed printer control codes in text, full justification series of commands, full "help" support, store common command series on disk for future use, etc. Easy "Set-Up" (for example, you just hit the key you want to use for a specific function, such as "cursor up", and the System reads an stores that key - no digging into tech manuals for codes, etc.); use supplied "set-ups", or remap the keyboard to what you are used too. Except for proportional printing, this package will DO IT ALL!

6800 or 6809 FLEX or SSB DOS, OS-9 - $175.00

Great Plains Computer Co.

STYLOGRAPH

A full-screen oriented WORD PROCESSOR -- (now runs on the Data-Comp and FHL Color FLEX Systems; uses the 51 x 24 Display Screens). Full screen display and editing (i.e., what you see is what you get); supports the Daisy Wheel proportional printers.

SPECIAL CCF - $195.00

F and O - $295.00 U - $395.00

SPELL

Fast Computer Dictionary.

F, CCF, OS/9 - $125.00 U - $175.00

MAIL MERGE

Greatly extends the power and flexibility of STYLOGRAPH.

F, CCF, O - $145.00 U - $195.00

VISA MasterCard

TOLL FREE
1-800-338-6800
For Ordering

SOUTH EAST MEDIA

5900 Cassandra Smith Rd.
Hixson, TN 37343

info (615) 842-4601

CoCo OS-9™ FLEX™
SOFTWARE

Availability Legends —
F = FLEX, CCF = Color Computer FLEX
O = OS-9, CCO = Color Computer OS-9
U = UniFLEX
CCD = Color Computer Disk
CCT = Color Computer Tape

™FLEX is a trademark of Technical Systems Consultants
™OS9 is a trademark of Microware

Great Plains Computer Co.

MAIL MERGE

Greatly extends the power and flexibility of STYLOGRAPH. Allows Multiple Text files to be printed out as one large document. Provides for merging information into the Text File during printing (such as different names and addresses), etc.

F, CCF, O - $145.00
U - $195.00

Southeast Media

SPELL8 "Computer Dictionary"

OVER 120,000 words!

No more "Let your fingers do the walking through the Dictionary" while you are entering Text with your favorite Editor or Word Processor. SPELL8 is more than just "another Spelling Checker"; it allows you to look up a word from within your Editor or Word Processor so that you KNOW it is right WHEN YOU TYPE IT IN with the SPN.CMD Utility (which operates in the FLEX Utility Space). Yes, it ALSO allows you to check and update the Text after you are finished; along with allowing you to ADD WORDS to the Dictionary, "Flag" questionable words in the Text for evaluation later, "View a word in context" before changing or ignoring, etc. SPELL8 first checks a "Common Word Dictionary", then the normal Dictionary, then a "Personal Word List", and finally, any "Special Word List" you may have specified. SPELL8 also allows the use of Small Disk Storage systems.

F and CCF - $129.95

Great Plains Computer Co.

SPELL

Fast Computer Dictionary -- allows directly changing the Text File, adding words to the dictionary, etc. 75,000 words in less than 400 sectors.

F, CCF, OS/9 - $125.00
U - $175.00

DATA BASE MANAGEMENT SYSTEMS

Intercomputer Applied Business Systems

XDMS

Possibly one of the most powerful Database Management Systems' available, this machine language program is small enough to operate on a **single sided 5" disk**, yet provides the **speed of** M.L. and **power** limited only by the user's imagination. This DMS supports Relational, Sequential, Hierarchical, and Random Access File Structures, and has Virtual Memory capabilities for those Giant Data Bases. XDMS Level I provides a functional "entry level" System which provides for defining a Data Base, entering and changing the Data, and producing Reports. XDMS Level II adds the POWERFUL "GENERATE" facility which uses an English Language Command Structure in manipulating the Data to create new File Structures, Sort, Select, Calculate, etc. XDMS Level III adds several special "Utilities" which provide additional ease of working with the various structures, changing System Parameters, etc.

XDMS Lvl I - F & CCF - $129.95
XDMS Lvl II - F & CCF - $199.95
XDMS Lvl III - F & CCF - $269.95
XDMS System Manual only - $24.95

Great Plains Computer Co.

INFOMG DBMS

An XBASIC, Menu Driven, DBMS with "Built-In" Audit Tracking, Extremely Powerful Report & Format Capabilities, etc. This **Time** Proven DBMS will become the "Work Horse" of your Software Stable.

F and CCF $295.00
U $395.00

ACCOUNTING PACKAGES

Great Plains Computer Co. and **Universal Data Research, Inc.** both have Business Packages written in TSC XBASIC for FLEX, **CoCo FLEX**, and **UniFLEX** ----

• • • • • • • • • • • • •

- - - - Call 800-338-6800 for more information - - - -

• • • • • • • • • • • • •

TOLL FREE
1-800-338-6800
For Ordering

SOUTH EAST MEDIA

5900 Cassandra Smith Rd.
Hixson, TN 37343
for information
call (615) 842-4601

CoCo OS-9™ FLEX™
SOFTWARE

Computer Systems Consultants

BASIC UTILITY PROGRAMS

Ten BASIC Programs to:

A BASIC Resequencer with EXTRAs over "RENUM"; works with ALL Versions of FLEX BASIC **AND** the Precompiler, checks for missing label definitions, processes Disk to Disk instead of in Memory.

Compare, Merge, or **Generate Updates** between two BASIC Programs, check BASIC Sequence Numbers, compare two unsequenced files, and 5 Programs for establishing a Master Directory of several Disks, and sorting, selecting, updating, and printing paginated listings of these files.

A BASIC Cross-Reference Program, written in Assembly Language, which provides an X-Ref Listing of the Variables and Reserved Words in TSC BASIC, XBASIC, and PRECOMPILER BASIC Pro rams.

ALL Utilities include Source (either BASIC or Source Code). An EXCELLENT Value!

F and CCF - $25.00
UniFLEX - $50.00

Computer Systems Consultants

FULL SCREEN INVENTORY/MRP

The Full Screen Inventory System provides a means of maintaining small inventories. Using a linked, keyed random file structure based upon the item field, it keeps the file in alphabetical order for easier inquiry. With the FIND command, the user may locate and/or print all records matching on partial or complete item, description, vendor, or attributes. Items in backorder or below minimum stock levels may be located and/or printed thru the same process. Printed output may be produced in item or vendor order. A materials requirement planning (MRP) capability for manufacturing environments is included to allow the maintenance and analysis of Hierarchical assemblies of items in the inventory file. It requires TSC's Extended BASIC.

F and CCF - $100.00, U - $150.00

BUSINESS FORECASTING

The Virginia Company ***Bizpack***

BIZPACK is used for storing accounting, numeric, and financial data which can then be used for planning, budgeting, forecasting, analyzing, etc. While "Electronic Spreadsheets" are extremely useful in many situations, BIZPACK excels in businesses where there are numerous expense columns, revenue sources, significant business indicators, large numbers, erratic week-to-week and month-to-month fluctuations, etc. BIZPACK helps determine statistical relationships, establish trend lines, "smooths" data via moving averages, analyze seasonal data, adjusts for inflation, lags data in Statistics or Column functions, plots data, etc. BIZPACK is oriented toward time series analysis of businesses. The Program displays information on the screen in Columns of Information with each Row conforming to a defined Period of Time (weeks, months, years, etc.), and is very easy to use (data is easy to enter, change, and modify; commands can be renamed to suit the users requirements; unlimited ability to create specialized commands using common BASIC Statements; etc.). Requires TSC's Extended BASIC.

F and CCF - $135.00
with Source - $250.00

*** SPECIAL ***
Purchase XBASIC and BIZPACK together for $221.50
-- a Savings of $13.50 --

VISA MasterCard

"FLEX is a trademark of Technical Systems Consultants
"OS9 is a trademark of Microware

TOLL FREE 1-800-338-6800 For Ordering
SOUTH EAST MEDIA
5900 Cassandra Smith Rd. Hixson, TN 37343
info (615) 842-4601
CoCo OS-9™ FLEX™ SOFTWARE

Availability Legends —
F = FLEX, CCF = Color Computer FLEX
O = OS-9, CCO = Color Computer OS-9
U = UniFLEX
CCD = Color Computer Disk
CCT = Color Computer Tape

TOLL FREE
1-800-338-6800
For Ordering

SOUTH EAST MICRO

5900 Cassandra Smith Rd.
Hixson, TN 37343
for information
call (615) 842-4601

CoCo OS-9™ FLEX™
SOFTWARE

*** SPECIAL ***
Purchase XBASIC and DISKPACK together for $221.50
— a Savings of $13.50 —

Computer Systems Consultants
TABULA RASA SPREADSHEET

TABULA RASA is similar to DESKTOP/PLAN and provides for the generation and maintenance of tabular computation schemes often used for analysis of business, sales, and economic scenarios. Its menu-driven user interface provides these capabilities even to those users with no programming experience. Its extensive report-generation capabilities allow the user to generate professional results with minimum effort. It requires TSC's **Extended BASIC.**

F and CCF - $100.00, U - $125.00

Computer Systems Center
DYNACALC

THE Electronic Spread Sheet for 6809 Computer Systems. An extremely POWERFUL Business Tool, this Program will find an unlimited number of "non-business" applications, also (for example, a Full Junior College Electronics Curriculum was set up using **DYNACALC**). Advanced features like "Table Lookup" make Income Tax work easy; Column or Row Sorting for numerous applications; etc. Completely "Memory Resident", Machine Language, this Program is **FAST.** Provides STANDARD FLEX Text File output for use with BASIC, Word Processors, Pascal, "C", etc. Also available for Data-Comp and FHL FLEX systems using the 50 x 24 Displays.

F and SPECIAL CCF - $200.00
U - $395.00

ODDS & ENDS

Computer Systems Consultants
FULL SCREEN FORMS DISPLAY

This package supports any Serial Terminal with cursor control of Memory-Mapped Video Displays. The package substantially extends the screen input/output capabilities of TSC's Extended **BASIC** programs by providing a simple, table-driven method of describing and using full screen displays. These table entries are easy to set up and maintain, and are normally stored on disk and read as required. A simple, interactive means of generating the forms and the data field definitions is provided.

F and CCF - $50.00, U - $75.00

Computer Systems Consultants
FULL SCREEN MAILING LIST

The Full Screen Mailing List System provides a means of maintaining simple mailing lists. Using a random fill structure based on the first character of the name field, it maintains the file in alphabetical order for easier inquiry. With the FIND command, the user may locate all records matching on partial or complete name, city, state, zip, or attributes. Printed listings and output to labels may also be produced on the same selective basis. It requires TSC's Extended BASIC.

F and CCF - $100.00, U - $110.00

COLOR COMPUTER SOFTWARE

Stearns Electronics
FORTH

Intrigued by FORTH?? Here is a FORTH package **tailored** to the Color Computer! This package is supplied on Tape, with instructions for transferring it to **disk** if you wish. Written primarily in machine language, it's **speed is unparalleled.** A full Semigraphic-8 Editor is provided, along with "goodies" like Graphics and Sound Commands, Printer Commands, Auto-Repeat and Control Keys, etc. If you are interested in learning FORTH, a **Trace Feature** is provided which is invaluable. If you are a FORTH Pro, this package provides CPU carry Flag accessibility, Fast Task Multiplexing, Clean Interrupt Handling, etc. (Or, you won't "out grow" the Basic capabilities of this implementation). Combine this package with Leo Brodie's EXCELLENT Book "Starting FORTH", and you will be a FORTH Expert before you know it (and have a lot of fun doing it!).

Color Computer TAPE - $58.95

Custom Software Engineering, Inc.
Color Computer **GRAPHIC SCREEN PRINT** Programs

Dumps any "PMODE" Screen to the Printer with the BASIC USR Function. Shift the Printout Left or Right or Reverse Print (Dark for Light Screen and Vice Versa). All Programs on Tape.

GSPR	for R.S. LP-VII/VIII & DMP 100/200/400	$7.95
GSPE	for Epson w/ Graftrax and Graftrax +	$9.95
GSPG	for Gemini 10 and 15	$9.95
GSPP	for the Prowriter Printers	$9.95

Custom Software Engineering, Inc.
DATE-O-BASE CALENDAR Program

A Menu Driven **EXTENDED BASIC** Program which allows the entry of up to 12 Memos per Day, each of which may contain up to 28 Characters, for any day of the Month between the years 1700 and 2099. A **Graphic Calendar** shows which days contain Memos, and a "Key Word" Search is provided which can be output to the Screen or Printer.

TAPE DATE-O-BASE CALENDAR
(Each Tape File will hold up to 400 Memos) $16.95
DISK DATE-O-BASE CALENDAR
(4,000 Memos at 300/Month per Disk) $19.95

Custom Software Engineering, Inc.
That's INTEREST-ing

Interested in INTEREST (the Money Kind)? An EXTENDED BASIC Program that will help you deal with numerous problems requiring interest calculations. Present Value, Rate of Return, Current Bond Yield and Rate of Return to maturity, Loan Repayment Amortization Schedules, etc.

TAPE - $29.95

Custom Software Engineering, Inc.
DISK DATA HANDLER 64K

An EXTENDED BASIC Data Management System w/ Mach. Lang. Routines. Allows a max of 246 Chars. and 14 Fields per Record, and another Record can be linked to the first; 8 Char. Field Names, up to 99 Chars. per Field. Powerful On-Screen editor for input and update, flexible Output capabilities including output to Disk Files for use by other Programs. Change File Definition without re-entering the Data. Split Files, etc. Allows Multiple Field Sorts, Select on any combination of Fields, etc. An extremely POWERFUL TOOL! Instructions provide examples of Mailing Lists and a Financial Stock Profit and Loss Tracking System.

DISK - $54.95

Custom Software Engineering, Inc.
DISK DOUBLE ENTRY

DISK EXTENDED BASIC Accounting Program w/ Mach. Lang. Routines. A "Traditional" Accounting Package for Small Business, Clubs, Churches, Personal Use, etc. Up to four levels of subtotals with Trial Balance, Income Statement, and Balance Sheet Reports. DDE allows up to 300 accounts and a Trial Balance of $9,999,999.99. Transactions may be up to 14 lines long, and comments and explanations may be freely used. Accounts are traceable to the journal transaction, which may include comments. Screen reports allow review of past transactions and current balances.

DISK - $44.95

VISA MasterCard

*FLEX is a trademark of Technical Systems Consultants
*OS9 is a trademark of Microware

TOLL FREE
1-800-338-6800
For Ordering
SOUTH EAST MICRO
5900 Cassandra Smith Rd.
Hixson, TN 37343
Info (615) 842-4601
CoCo OS-9™ FLEX™
SOFTWARE

Availability Legend —
F = FLEX, CCF = Color Computer FLEX
O = OS-9, CCO = Color Computer OS-9
U = UniFLEX
CCD = Color Computer Disk
CCT = Color Computer Tape

Computer Systems Center

DYNASHARE

-- Multi-User, Multi-Tasking with FLEX --

Southeast Media is now shipping DYNASHARE **FROM STOCK** - the multi-user, multi-tasking capability of **DYNASHARE** allows FLEX users the advantages of more sophisticated and time saving computer usage without having to buy or learn a new language or Operating System syntax. **DYNASHARE**, as its name implies, allows true "time-sharing" operation under the popular FLEX operating system, and also allows each user to run two simultaneous jobs (multi-tasking); even on single-user systems. For example, while in EDIT, you can list another file or examine a directory. Or, you might look up an item in a Data Base while a Sort is in progress! **DYNASHARE** also provides some fringe benefits that will be greatly appreciated by FLEX users, including type-ahead, command line editing, and instant response to "escape".

DYNASHARE is the painless method! Use your existing Flex computer by simply adding 64K of RAM for each user and/or task. Fact is, you still use FLEX just like you always have! **DYNASHARE** is not intended as competition to UniFLEX. It does not improve on the speed of FLEX, and does not offer password protection or other niceties of a full-blown multi-user system. What **DYNASHARE does do** is give FLEX users a **low-cost way to use existing software in a multi-user, multi-tasking environment**, so your existing FLEX versions of BASIC, XBASIC, editors, assemblers, disassemblers, sort/merge packages, word processors, compilers, DYNACALC spread-sheet package, and so on are still good.

NOTE -- The **initial release** of DYNASHARE is for SWTPC 6/09 Computers, but versions will also be available for other popular extended-memory (up to **512K**) systems, such as HELIX and GIMIX. A minimum of **128K** of **RAM** will be required with **ALL** versions. **DYNASHARE** requires **64K of RAM for each active task**; thus a 256k system could allow foreground-background operation on two terminals, or foreground-only operation on four terminals.

AVAILABLE NOW from **Southeast Media** - **$200.00**

AUTHORS - PROGRAMMERS
QUALITY SOFTWARE NEEDED
FLEX - UniFLEX - OS/9 - Color Computer

For the past several months, we at the **Southeast Media Division** of **Computer Publishing, Inc.** (CPI), the parent company of '68' **MICRO JOURNAL** and **COLOR MICRO JOURNAL**, have debated expanding our software distribution business. Many other magazines have been doing so for years (in fact, MOST were in the Software Distribution Business BEFORE they began to publish a Magazine). Presently there are many fine examples of software that has been developed by YOU, our readers, that will never see the "light of day" due to the **Cost of Advertising** and **TIME and Cost** involved in the **production, distribution,** and **Customer SUPPORT** of that software unless SOMEONE, with enough exposure and the willingness to continually advertise, runs with the ball.

Software is the "backbone" for the REAL utilization of any Computer System, and ours are no exception! This has been no simple decision. While we realize that there could be some conflict with some of our advertisers, we ALSO hear a LOUD and CONTINOUS cry for **HELP** from our Readers. From day one, the **foremost concern** of '68' MICRO JOURNAL has been it's **READERS!** Therefore, our **Southeast Media Division** will accept, for appraisal for possible Distribution, 6809 software; Games, Utilities, Software Development, Business Application Programs, etc.

In the past there has been **too much** software offered that was not quite ready. We will strive to eliminate that element. But, right up front, we tell you only that we will do our very best; nothing more. Also, we will strive to keep cost to a bare minimum, while securing for the author a fair return in royalty payments, promptly paid, and in customer support for his product.

Of course, we will expect, no -- **DEMAND**, that the author keep the product free of errors (bugs), and maintain it in a prompt and business like manner. Also we shall require that authors be willing to furnish 'source' for those programs that justify, by price and utility, inclusion of same. The lack of source code, properly commented, is a continual complaint we hear. Not all programs will be sold with source, but where necessary, we will insist that it be included.

In some instances the program may be small or short and not justify itself as a "single" sale product. In this event it will be combined with other like programs, and offered as a package. In that event, the royalties will be split between the various authors.

If you have software that you feel will qualify under this program, please contact one of the people below. Remember, if your software has any problems or "funnies" -- **GET IT STRAIGHT BEFORE YOU CONTACT US!!** Also get your source code in proper shape and well commented; there is too much 99% code already drifting around.

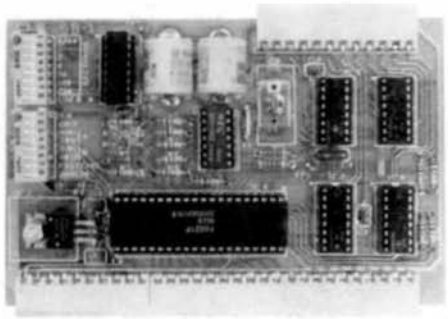
If your **software is READY** contact:
Bob May, Don Williams, or Tom Williams

Southeast Media is a division
of **Computer Publishing, Inc. (CPI)**,
a family of **100%** 68XX support facilities.

*FLEX is a trademark of Technical Systems Consultants
"OS9 is a trademark of Microware

Availability Legends —
F = FLEX, CCF = Color Computer FLEX
O = OS-9, CCO = Color Computer OS-9
U = UniFLEX
CCD = Color Computer Disk
CCT = Color Computer Tape

CALENDAR-CLOCK / TIMER / PARALLEL PORT



Calendar - Clock CLK68-1

- Keeps date and time whether or not the computer is on
- All clock functions software controlled
- No field accessible jumpers/pots are required - already set for access
- Day of week, month/day/year, hour/minute/sec (12/24 hr)

Interval Timer

- For printer spooling, multi-tasking, etc.
- Compatible with 05-9** and Flex 2/9*
- 05-9 runs faster with CLK68-1 than with timers such as SVTPC MP-T
- Generates interrupt approximately every 7ms although set to 25m sec.

Parallel I/O Port -- Fully buffered 8 bit parallel port

- Bit accessible select codes or output buffering for jumpers of one board
- Compatible with parallel printer software in many versions of BASIC

Construction -- Solder masked, silver coated, & with screws

Manual -- Well documented - 36 pages

Assembled and tested $119.95 Kit $89.95
Goldplated bus conn 7.50 2 MHz option 2.50
Disk 5 or 8 in. SS8 or Flex* OS-9 Available NOW 14.95

** 05-9 is a trademark of Microware Systems Corporation
* Flex is a trademark of Technical Systems Consultants, Inc.

ROBERTSON ELECTRONICS
1003 Warm Sands Dr. SE
Albuquerque, NM 87123

Phone (505) 294-0025
NM residents add 4% tax
Add $3 Shipping & Handling

COMPARE

our EPROM PROGRAMMER with the field.

All data taken directly from manufacturer's current advertising. Software, interfaces, or personality modules may also be required at additional cost.

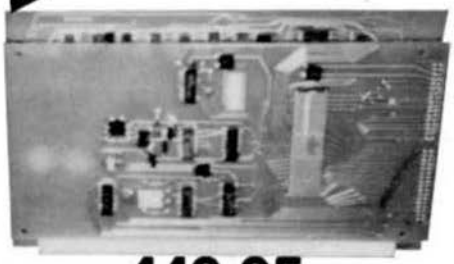
- Triple voltage EPROM
- Supplied in kit form

	A	B	C	D	E	F
INTERFACE	S30	PAR	PAR	SER	S30	SER	SER
INTELLIGENT	NO	NO	NO	YES	NO	YES	YES
PROGRAMS							
2704*			●				●
2508	●			●	●	●	
2708*			●		●		●
2758	●	●	●	●		●	●
2516	●	●	●	●	●	●	
2716	●	●	●				●
2716*			●		●		●
2532	●		●	●	●	●	●
2732	●		●	●	●	●	●
2732A	●		●		●	●	●
2564	●		●		●	●	●
2764	●		●		●	●	●
2528	●		●		●		
27128	●						
2816							●
68764			●				
8748						●	
8749						●	
TOTAL	11	3	12	6	11	11	11
PRICE	$125	$45*	$169	$289	$375	$489	$575

EPROM EPROM Programmer, $125. Personality module for 2508, 2758, 2516, and 2716 included. Specify CPU, disk size, and operating system (TSC's FLEX or 6809's DOS) when ordering. Manual only, $10; refundable with EPROM purchase.

UNITEK • P.O. Box 671 • Emporia, VA 23847

SUPER CPU . . . only from LSI **68008**



449.95 KIT

We're mighty proud of our new processor card. We're giving you the ability to go 68000 without major changes to your system. Our new CPU gives you these advanced features:

- Dynamic partitioning memory management unit with bound check register.
- On-board timer for multi-user/multi-tasking applications.
- On-board boot-strap EPROM and Monitor EPROM space.
- Vectored priority interrupt generator.
- On-board wait state generator.
- User selectable bus options that includes a new higher bandwidth bus mode.
- And many more.

S68K/08-CPU-ASSEM. & TESTED 549.95
S68K/08-CPU-KIT FORM 449.95
(KIT INCLUDES PROCESSOR, CRYSTAL, SOCKETS AND CONNECTORS)
DISK CONTROLLER SU PORTED DC3, DC4, DMF2, SDC8.

Announcing . . .

THE *SHELL* FOR FLEX 9™ *

We are pleased to announce the SHELL, a UNIX ** like shell that supports I/O redirection, pipes, macro substitution and programmable shell scripts! The shell will work with all your existing programs and utilities. Requires 56K of user ram, FLEX 9™ * version 2.8 and above. The shell occupies the top 8K of user ram. An excellent tool for the 6809 community.

FLX/SHO9-8 . . . 8 inch version 90.00
FLX/SHO9-5 . . . 5.25 inch version . . . 90.00
ONE YEAR MAINTENANCE 22.50

LSI STANDS BEHIND ITS PRODUCTS
1 YEAR LIMITED WARRANTY
ON ALL OUR PRODUCTS

ANNOUNCING THE LSI 68000 USERS GROUP

Join the Group by sending us your name and address. You will receive our monthly publication with free public domain user programs and software updates.
New members of the users group will receive a $30.00 discount on THEIR FIRST LSI hardware purchase. Anyone that donates a program to the group will receive our current user group software on a formatted CP/M readable disk.

256K RAM CARD
Using the latest LSI technology this 256K RAM CARD makes a perfect addition to your S68K system. Uses MRDY for refresh arbitration.
S68K/08-256K . . . $750.00

SDC8 CONTROLLER
A SS50 DMA disk controller for use with either 68000 bus modes or 6809 bus modes. Features a high reliability digital data separator. (No analog circuits to drift) and full 1 Megabyte addressing range.
S68K/08-SDC8-$550.00

CP/M-68K +
DRI's famous CP/M made for the 68000. It includes a C compiler, relocatable assembler, linking loader, librarian and many utilities. It is source compatible with all other CP/M operating systems.
CP/M-68K + . . . $350.00

N.Y. residents add sales tax.

VISA master charge

and COD accepted

*FLEX 9™ is a trademark of Technical Systems Consultants Inc.
+ CP/M-68K is a registered trademark of Digital Research, Inc.
** UNIX is a registered trademark of Bell Labs.
All prices and offers subject to change without notice.

LSI Enterprises Ltd.
PO Box 1227
Woodhaven, NY 11421
(212) 423-5596

TEN MOST-ASKED QUESTIONS about DYNACALC™

THE ELECTRONIC SPREAD-SHEET FOR 6809 COMPUTERS

1. What is an electronic spread-sheet, anyway?
Business people use spread-sheets to organize columns and rows of figures. DYNACALC simulates the operation of a spread-sheet without the mess of paper and pencil. Of course, corrections and changes are a snap. Changing any entered value causes the whole spread-sheet to be re-calculated based on the new constants. This means that you can play, 'what if?' to your heart's content.

2. Is DYNACALC just for accountants, then?
Not at all. DYNACALC can be used for just about any type of job. Not only numbers, but alphanumeric messages can be handled. Engineers and other technical users will love DYNACALC's sixteen-digit math and built-in scientific functions. You can build worksheets as large as 256 columns or 256 rows. There's even a built-in sort command, so you can use DYNACALC to manage small data bases — up to 256 records.

3. What will DYNACALC do for ME?
That's a good question. Basically the answer is that DYNACALC will let your computer do just about anything you can imagine. Ask your friends who have VisiCalc™, or a similar program, just how useful an electronic spread-sheet program can be for all types of household, business, engineering, and scientific applications. Typical uses include financial planning and budgeting, sales records, bills of material, depreciation schedules, student grade records, job costing, income tax preparation, checkbook balancing, parts inventories, and payroll. But there is no limit to what YOU can do with DYNACALC.

4. Do I have to learn computer programming?
NO! DYNACALC is designed to be used by non-programmers, but even a Ph.D. in Computer Science can understand it. Even experienced programmers can get jobs done many times faster with DYNACALC, compared to conventional programming. Built-in HELP messages are provided for quick reference to operating instructions.

5. Do I have to modify my system to use DYNACALC?
Nope. DYNACALC uses any standard 6809 configuration, so you don't have to spend money on another CPU board or waste time learning another operating system.

Order your DYNACALC today!

Foreign Dealers:

Australia & Southeast Asia: order from Paris Radio Electronics, 161 Bunnerong Road (PO Box 380) Kingsford, 2032 NSW Australia. Telephone: 02-344-9111.

United Kingdom: order from Compusense, Ltd., PO Box 169, London N13 4HT. Telephone: 01-882-0681.

Scandinavia: order from Swedish Electronics hk AB, Murargatan 23-25, Uppsala 5-754 37 Sweden. Telephone: 18-25-30-00.

6. Will DYNACALC read my existing data files?
You bet! DYNACALC has a beautifully simple method of reading and writing data files, so you can communicate both ways with other programs on your system, such as the Text Editor, Text Processor, Sort/Merge, STYLOGRAPH™ word processor, RMS™ data base system, or other programs written in BASIC, C, PASCAL, FORTRAN, and so on.

7. How fast is DYNACALC?
Very. Except for a few seldom-used commands, DYNACALC is memory-resident, so there is little disk I/O to slow things down. The whole data array (worksheet) is in memory, so access to any point is instantaneous. DYNACALC is 100% 6809 machine code for blistering speed.

8. Is there a version of DYNACALC for MY system?
Probably. You need a 6809 computer (32k minimum) with FLEX™, UniFLEX™, or OS-9™ operating system. You also need a decent crt terminal, one with at least 80 characters per line, and direct cursor addressing. If your terminal isn't smart enough for DYNACALC, you probably need a new one anyway. The UniFLEX and OS-9 versions of DYNACALC allow you to mix different brands of terminal on the same system. There's also a special version of DYNACALC for Color Computers equipped with FLEX (Frank Hogg or Data-Comp versions).

9. How much does DYNACALC cost?
The FLEX versions are just $200 per copy; UniFLEX version $395; OS-9 version (works with LEVEL ONE or LEVEL TWO) $250. Orders outside North America add $7 per copy for postage. We encourage dealers to handle DYNACALC, since it's a product that sells instantly upon demonstration. Call or write on your company letterhead for more information.

10. Where do I order DYNACALC?
See your local DYNACALC dealer, or order directly from CSC at the address below. We accept telephone orders from 10 am to 6 pm, Monday through Friday. Call us at 314-576-5020. Your VISA or MasterCard is welcome. Please specify diskette size for FLEX or OS-9 versions. Software serial number is required for the UniFLEX version.

Computer Systems Center
13461 Olive Blvd.
Chesterfield, MO 63017
(314) 576-5020

UniFLEX software prices include maintenance for the first year.

DYNACALC is a trademark of Computer Systems Center

VisiCalc is a trademark of VisiCorp.
STYLOGRAPH is a trademark of Great Plains Computer Co.
RMS is a trademark of Washington Computer Services.
FLEX and UniFLEX are trademarks of TSC.
OS-9 is a trademark of Microware and Moto ola.

DYNAMITE+™

"THE CODE BUSTER"

disassembles any 6809 or 6800
machine code program into beautiful source

- Learn to program like the experts!
- Adapt existing programs to your needs!
- Convert your 6800 programs to 6809!
- Automatic LABEL generation.
- Allows specifying FCB's, FCC's, FDB's, etc.
- Constants input from DISK or CONSOLE.
- Automatically uses system variable NAMES.
- Output to console, printer, or disk file.
- Available for all popular 6809 operating systems.

FLEX™ $100 per copy; specify 5" or 8" diskette.
OS-9™ $150 per copy; specify 5" or 8" diskette.
UniFLEX™ $300 per copy; 8" diskette only.

For a free sample disassembly that'll convince you DYNAMITE + is the world's best disassembler, send us your name, address, and the name of your operating system.

Order your DYNAMITE+ today!

See your local DYNAMITE + dealer, or order directly from CSC at the address below. We accept telephone orders from 10 am to 6 pm, Monday through Friday. Call us at 314-576-5020. Your VISA or MasterCard is welcome. Orders outside North America add $5 per copy. Please specify diskette size for FLEX or OS-9 versions.

Foreign Dealers:

Australia & Southeast Asia: order from Paris Radio Electronics, 161 Bunnerong Road (PO Box 380) Kingsford, 2032 NSW Australia. Telephone: 02-344-9111.

United Kingdom: order from Compusense, Ltd., PO Box 169, London N13 4HT. Telephone: 01-882-0681.

Scandinavia: order from Swedish Electronics hk AB, Murargatan 23-25, Uppsala S-754 37 Sweden. Telephone: 18-25-30-00.

Computer Systems Center
13461 Olive Blvd.
Chesterfield, MO 63017
(314) 576-5020

CSC

UniFLEX software prices include maintenance for the first year.

DYNAMITE + is a trademark of Computer Systems Center.

FLEX and UniFLEX are trademarks of TSC.
OS-9 is a trademark of Microware and Motorola.

Dealer inquiries welcome.

'68' MICRO JOURNAL

★ The only ALL 6800 Computer Magazine.

★ More 6800 material than all the others combined:

MAGAZINE COMPARISON
(2 years)
Monthly Averages

	6800 Articles			TOTAL
KB	BYTE	CC	DOBB'S	PAGES
7.8	6.4	2.7	2.2	19.1 ea. mo.

Average cost for all four each month: $6.53
(Based on advertised 1-year subscription price)

68' cost per month: $2.04

That's Right! Much, Much More

for About

1/3 the Cost!

OK, PLEASE ENTER MY SUBSCRIPTION

Bill My: Master Charge ☐ — VISA ☐

Card # _____ Exp. Date_____

For ☐ 1-Year ☐ 2 Years ☐ 3 Years

Enclosed: $_____

Name_____

Street_____

City_____ State_____ Zip_____

My Computer Is:_____

68 Micro Journal
5900 Cassandra Smith Rd.
Hixson, TN 37343

SUBSCRIPTION RATES

USA
1 Year $24.50, 2 Year $42.50, 3 Year $64.50

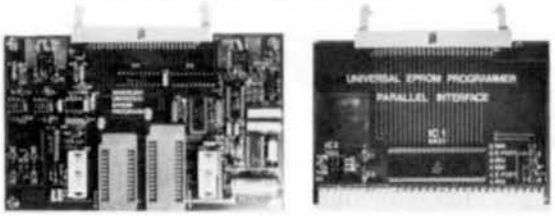
*FOREIGN SURFACE Add $12.00 per Year to USA Price

*FOREIGN AIRMAIL Add $36.00 per Year to USA Price

**CANADA & MEXICO Add $5.50 per Year to USA Price
Cash (USA) or drawn on a USA Bank!!!

WINDRUSH MICRO SYSTEMS

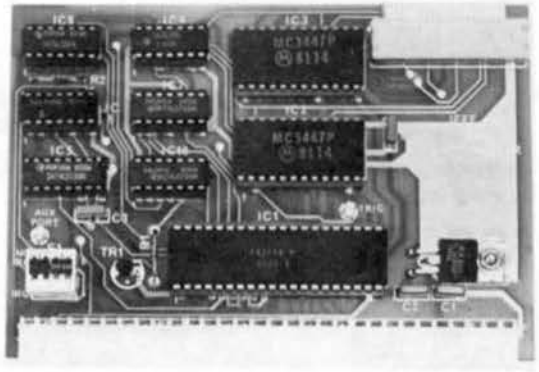
UNIVERSAL EPROM PROGRAMMER



- PROGRAMS and VERIFIES 2508, 2708, 2516, 2716, 2532, 2732A, 2564, and 2764 EPROMS. Minor hardware mods are required to program the INTEL 27128.
- Tri-volt and Single Volt 2508/2708 and 2516/2716 devices are supported.
- ZIF sockets with mode selector switches eliminate 'personality modules'.
- Twin boards with five feet of twisted pair planar cable puts the programmer out on the bench where it belongs.
- SS-30 and EXORCISOR interfaces are available.
- Menu driven software provides the following facilities:
  a. MOVE blocks of memory within the buffer.
  b. READ an EPROM into the buffer.
  c. VERIFY an EPROM against the buffer.
  d. EXAMINE and change the contents of the buffer.
  e. DUMP the contents of the buffer in HEX and ASCII.
  f. FILL a selected area of the buffer with a specified character.
- Software available for all versions of SSB DOS, FLEX 2, FLEX 9 and OS-9. Assembly language source files supplied on disk....enables customising.
- Well documented users manual provides step-by-step adaptation and operating instructions.

AVAILABLE FROM GIMIX IN THE U.S.A.

IEEE-488



- SUPPORTS ALL PRINCIPAL MODES OF THE IEEE-488 (1975/8) BUS SPECIFICATION:
  - Talker
  - Listener
  - System Controller
  - Serial Poll
  - Parallel Poll
  - Group Trigger
  - Single or Dual Primary Address
  - Secondary Address
  - Talk only...Listen only
- Fully documented with a complete reprint of the KILOBAUD article on the IEEE bus.
- Low level assembly language drivers suitable for 6800, 6801, 6802, 6803, 6808 and 6809 are supplied in the form of listings. These drivers have been extensively tested and are GUARANTEED to work!
- Single SS-30 board (4, 8, or 16 addresses per port), fully socketed, gold plated bus connectors, and IEEE interface cable assembly.

PL/9

EDITOR/COMPILER/DE-BUGGER

- Friendly inter-active environment where you have INSTANT access to the Editor, the Compiler, and the Trace-Debugger, which, amongst other things, can single step the the program a SOURCE line at a time. You also have direct access to any FLEX utility and your System Monitor.
- 250 page manual is organized as a tutorial with plenty of examples.
- Fast single Pass compiler produces 8K of COMPACT and FAST 6809 machine code output per minute with no run-time overheads or license fees.
- Fully compatible with TSC text editor format disk files
- Signed and unsigned BYTEs and INTEGERs, 32-bit floating point REALs.
- Vectors (single dimension arrays) and Pointers are supported.
- Mathematical expressions: (+), (-), (*), (/), modulus (\), negation (-)
- Expression evaluators: (=), (<>), (<), (>), (>=), (<=)
- Bit operators: (AND), (OR), (EOR/XOR), (NOT), (SHIFT), (SWAP)
- Logical operators: (.AND), (.OR), (.EOR/.XOR).
- Control statements: IF..THEN..ELSE, IF..CASE?..CASE?..ELSE, BEGIN..END, WHILE.., REPEAT..UNTIL, REPEAT..FOREVER, CALL, JUMP, RETURN, BREAK, GOTO.
- Direct access to (ACCA), (ACCB), (ACCD), (CCR) and (XREG).
- FULLY supports the MC6809 SWI, SWI2, SWI3, NMI, FIRQ, IRQ and RESET vectors. Writing a self-starting (from power-up) program that uses ANY, or ALL, of the MC6809 interrupts is an absolute snap!
- Procedures may be passed and may return variables. This makes them functions which behave as though they were an integral part of PL/9.
- Several fully documented library function modules are supplied: IOSUBS, BITIO, MATHIO, MEMIO, FLEXIO, SCIPACK, STRSUBS, and REALCON.

'... THIS IS THE MOST EFFICIENT COMPILER I HAVE FOUND TO DATE.'

Quoted from Ron Anderson's FLEX User Notes column. Need we say more?

MACE/XMACE

A co-resident EDITOR/ASSEMBLER for the 6809 written by Graham Trott which takes most of the pain out of assembly language program development:

- Friendly inter-active environment where you have INSTANT access to the Editor, the Assembler, FLEX and your System Monitor.
- MACE can also produce ASMPROC's for PL/9 with the assembly language source passed to the output file as comments.
- Includes XMACE a co-resident 6800/1/2/3/8 EDITOR/CROSS ASSEMBLER.

'C'

This is the FLEX version of the James McCosh 'C' compiler that is also available on UNIFLEX from SWTP and OS-9 from Microware:

- The FLEX implementation supports the full Kernighan and Ritchie 'C' specification except 'floats', 'doubles', and 'bit-fields'.
- Produces very efficient assembly language source output with the 'C' source optionally interleaved as comments.
- Built-in optimizer will shorten object code by about 11%
- Supports interleaved assembly language programs.
- The TSC relocating assembler/linking loader (SPOO-17) is REQUIRED.

MACE	(6809 FLEX only)	$ 98.00
PL/9	(6809 FLEX only)...(a steal at this price!)	$198.00
'C'	(a 56K system and the TSC SPOO-17 package is req'd)	$295.00
IEEE-488	with IEEE-488 cable assembly	$298.00
SS-30	Universal EPROM programmer w/one version of software	$375.00
EXORciser	Universal EPROM programmer w/one version of software	$395.00
SOFTWARE	Drivers for a 2nd, 3rd, or 4th operating system	$ 25.00

ALL PRICES INCLUDE AIR MAIL POSTAGE

WE STOCK THE FOLLOWING COMPANIES PRODUCTS: GIMIX, SSB, FHL, MICROWARE, TSC, LUCIDATA, AND ALFORD & ASSOCIATES.

FLEX (tm) is a trademark of Technical Systems Consultants, OS-9 (tm) is a trademark of Microware Systems Corporation, MDOS (tm) and EXORciser (tm) are trademarks of Motorola Incorporated.

An SS-50C all CMOS 256K STATIC RAM board will be available SOON!

Write for details & pricing.

WORSTEAD LABORATORIES, NORTH WALSHAM, NORFOLK, ENGLAND. NR28 9SA.
TEL: (0692) 405189
TLX: 97360 SHARET G

FEATURES THE POWERFUL, THIRD GENERATION, MOTOROLA 6809 PROCESSOR!

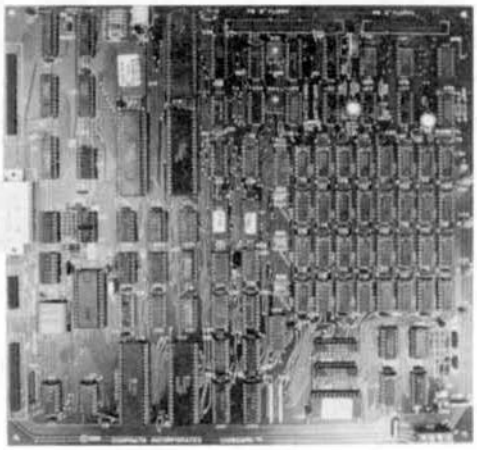
THE 6809 "UNIBOARD"™

SINGLE BOARD COMPUTER KIT

New!

PERFECT FOR COLLEGES, OEM'S, INDUSTRIAL AND SCIENTIFIC USES!

64K RAM! DOUBLE DENSITY FLOPPY DISK CONTROLLER!



BLANK PC BOARD
$99.95
WITH PAL'S, AND TWO EPROMS.
FOR 5-1/4 OR 8 INCH SOURCE DISKETTE ADD $10.

$399.00
COMPLETE KIT! FULLY SOCKETED.

ALL OPTIONS ARE STANDARD. NO EXTRAS TO BUY!

THE COMPACTA UNIBOARD™: Through special arrangement with COMPACTA INC., we are proud to have been selected the exclusive U.S. Mfg. of their new 6809 UNIBOARD™ COMPUTER KIT. Many software professionals feel that the 6809 features probably the most powerful instruction set available today on ANY 8 bit micro. Now, at last, all of that immense computing power is available at a truly unbelievably low price.

ALL SALES ARE MADE SUBJECT TO THE TERMS OF OUR 90 DAY LIMITED WARRANTY. A FREE COPY IS AVAILABLE UPON REQUEST.

FEATURES:

- ★ 64K RAM using 4116 RAMS.
- ★ 6809E Motorola CPU.
- ★ Double Density Floppy Disk Controller for either 5-1/4 or 8 Inch drives. Uses WD1793.
- ★ On board 80 x 24 video for a low cost console. Uses 2716 Char. Gen. Programmable Formats. Uses 6845 CRT Controller.
- ★ ASCII keyboard parallel input interface. (6522)
- ★ Serial I/O (6551) for RS232C or 20 MA loop.
- ★ Centronics compatible parallel printer interface. (6522)
- ★ Buss expansion interface with DMA channel. (6844)
- ★ Dual timer for real time clock application.
- ★ Powerful on board system monitor (2732). Features commands such as Go To, Alter, Fill, Move, Display, or Test Memory. Also Read and Write Sectors. Boot Normal, Unknown, and General Flex™.

YOUR CHOICE OF POPULAR DISK OPERATING SYSTEMS:

FLEX™ from TSC	$149
OS9™ from Microware	$199

Specify 5-1/4 or 8 Inch

PC BOARD IS DOUBLE SIDED, PLATED THRU SOLDER MASKED, 11 x 11-1/2 IN.

Digital Research Computers
(OF TEXAS)
P.O. BOX 461565 • GARLAND, TEXAS 75046 • (214) 271-3538

TERMS: Shipments will be made approximately 3 to 6 weeks after we receive your order. VISA, MC, cash accepted. Add $4.00 shipping. USA AND CANADA ONLY

64K SS-50 STATIC RAM

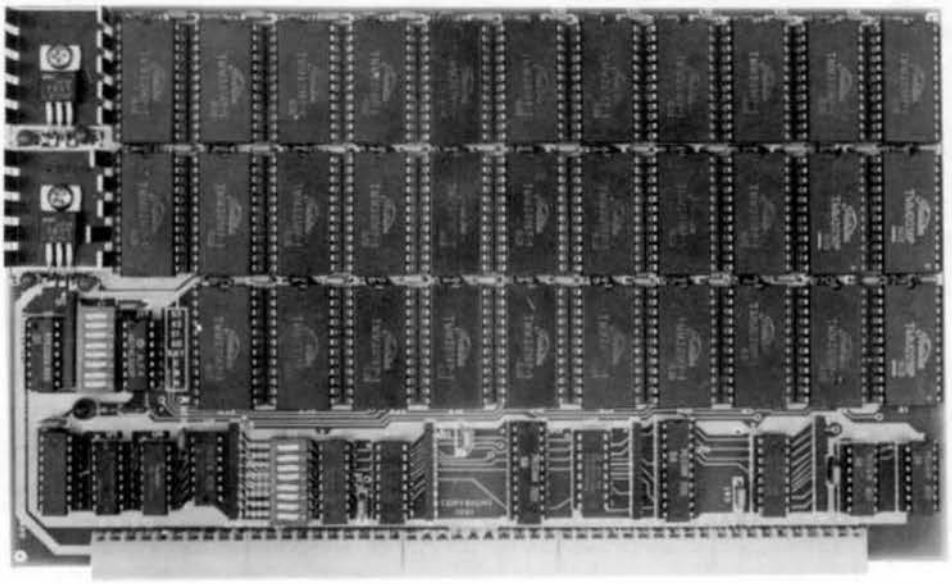
$179.00 (48K KIT)

NEW!

NEW!

LOW POWER!

RAM OR EPROM!



BLANK PC BOARD WITH DOCUMENTATION $52

SUPPORT ICs + CAPS -	$18.00
FULL SOCKET SET -	$15.00

ASSEMBLED AND TESTED **ADD $50**

FEATURES:

- ★ Uses new 2K x 8 (TMM 2016 or HM 6116) RAMs.
- ★ Fully supports Extended Addressing.
- ★ 64K draws only approximately 500 MA.
- ★ 200 NS RAMs are standard. (TOSHIBA makes TMM 2016s as fast as 100 NS. FOR YOUR HIGH SPEED APPLICATIONS.)
- ★ Board is configured as 3-16K blocks and 8-2K blocks (within any 64K block) for maximum flexibility.
- ★ 2716 EPROMs may be installed anywhere on Board.
- ★ Top 16K may be disabled in 2K blocks to avoid any I/O conflicts.
- ★ One Board supports both RAM and EPROM.
- ★ RAM supports 2MHZ operation at no extra charge!
- ★ Board may be partially populated in 16K increments.

56K	$219
64K	$249

CLOSE OUT SPECIAL

WE HAVE DROPPED OUR 32K SS-50 STATIC RAM BOARD WHICH USED 2114 LOW POWER RAMS. WE WILL SELL THE REMAINING STOCK OF BLANK PCB'S WITH DATA FOR $17.50 EA. THESE FORMERLY SOLDFOR $50.

16K STATIC RAMS?

The new 2K x 8, 24 PIN, static RAMs are the next generation of high density, high speed, low power, RAMs. Pioneered by such companies as HITACHI and TOSHIBA, and soon to be second sourced by most major U.S. manufacturers, these ultra low power parts, feature 2716 compatible pin out. Thus fully interchangeable ROM/RAM boards are at last a reality, and you get BLINDING speed and LOW power thrown in for virtually nothing.

Digital Research Computers
(OF TEXAS)
P.O. BOX 401565 • GARLAND, TEXAS 75040 • (214) 271-3538

TERMS: Add $2.00 postage. We pay balance. Order under $15 add 75¢ handling. No C.O.D. We accept Visa and MasterCharge. Tex. Res. add 5% Tax. Foreign orders (except Canada) add 20% P & H. Orders over $50, add 85¢ for insurance.

OS/9, FLEX, COCO, UNIFLEX SOFTWARE

SUPER SLEUTH DISASSEMBLER EACH $99-FLEX, $101-OS/9, $100-UNIFLEX
interactively generates source on disk with labels, includes xref, label definition, binary file editing, etc.
specify 6800,1,2,3,5,8,9/6502 version or Z-80/8080/85 version

(OBJECT ONLY) EACH $50-FLEX & OS/9, $49-COCO DOS
COCO DOS available in 6800,1,2,3,5,8,9/6502 version only

CROSS-ASSEMBLERS EACH $50-FLEX, $55-OS/9, $60-UNIFLEX, ALL $100
specify for 6800/1, 6502, 6805, Z-80, or 8080/48/85
OS/9 version requires Microware RMA or FHL OSM macro assembler
FLEX version requires TSC ASMB or FHL ASM or OSM macro assembler

DEBUGGING SIMULATORS EACH $75-FLEX, $100-OS/9, $80-UNIFLEX
specify 6800/1, 6805/146805, 6502, or (6809 OS/9 only)

OS/9 AND UNIFLEX SIMULATORS EACH $100-FLEX
debug OS/9 and UNIFLEX application programs under FLEX using TSC DEBUG facility

6502 TO 6809 ASSEMBLER TRANSLATOR $75-FLEX, $85-OS/9, $80-UNIFLEX
translates 6502 programs to 6809, noting inexact conversions

6800 TO 6809 & 6809 PIC TRANSLATORS $50-FLEX, $75-OS/9, $60-UNIFLEX
translates 6800 programs to 6809, 6809 programs to PIC

FULL-SCREEN FLEX AND UNIFLEX TSC XBASIC PROGRAMS FOR 6809
(with complete cursor control)

DISPLAY GENERATOR/DOCUMENTOR $50-FLEX, $75-UNIFLEX
MAILING LIST SYSTEM $100-FLEX, $110-UNIFLEX
INVENTORY WITH MRP $100-FLEX, $110-UNIFLEX
TABULA RASA SPREADSHEET $100-FLEX, $120-UNIFLEX

DISK UTILITY PROGRAM LIBRARY $50-FLEX
edit disk sectors, sort directory, maintain master catalog, etc.

CMODEM TELECOMMUNICATIONS PROGRAM $50-FLEX & OS/9 & UNIFLEX
provides menu-driven telecommunications facilities, with terminal mode, up/down load, MODEM7 protocol, etc.

5.25" SOFT-SECTORED DISKETTES EACH SET OF 50 $75-SSDD, $85-DSDD
with Tyvek jackets, hub rings, labels

Specify operating system, computer make and type, terminal type
Programs provided in source form on diskette: specify size and density
Contact CSC for full catalog and dealer info; printed manuals provided with products
For VISA and MASTER CARD, give account, exp. date, phone. US funds only.
Add 5% for shipping software, but not for diskettes.
(UNI)FLEX trademark Technical Systems Consultants. OS/9 trademark Microware.

Computer Systems Consultants, Inc.
1454 Latta Lane, Conyers, GA 30207
Telephone Number 404-483-1717/4570

SOFTWARE FOR THE HARDCORE

** FORTH PROGRAMMING TOOLS from the 68XX&X **
** FORTH specialists — get the best!! **

NOW AVAILABLE — A variety of rom and disk FORTH systems to run on and/or do TARGET COMPILATION for

6800, 6301/6801, 6809, 68000, 8080, Z80

Write or call for information on a special system to fit your requirement.

Standard systems available for these hardware—

EPSON HX-20 rom system and target compiler
6809 rom systems for SS-50, EXORCISER, STD, ETC.
COLOR COMPUTER
6800/6809 FLEX or EXORCISER disk systems.
68000 rom based systems
68000 CP/M-68K disk systems, MODEL II/12/16

tFORTH is a refined version of FORTH Interest Group standard FORTH, faster than FIG-FORTH. FORTH is both a compiler and an interpreter. It executes orders of magnitudes faster than interpretive BASIC. MORE IMPORTANT, CODE DEVELOPMENT AND TESTING is much, much faster than complied languages such as PASCAL and C. If Software DEVELOPMENT COSTS are an important concern for yo , you need FORTH!

firmFORTH™ is for the programmer who needs to squeeze the most into roms. It is a professional programmer's tool for compact rommable code for controller applications.

* tFORTH and firmFORTH are trademarks of Talbot Microsystems
* FLEX is a trademark of Technical Systems Consultants, Inc.
* CP/M-68K is trademark of Digital Research, Inc.

tFORTH℠
from TALBOT MICROSYSTEMS
NEW SYSTEMS FOR
6301/6801, 6809, and 68000

---> tFORTH SYSTEMS <---

For all FLEX systems: GIMIX, SWTP, SSB, or EXORcisor Specify 5 or 8 inch diskette, hardware type, and 6800 or 6809.

** tFORTH — extended fig FORTH (1 disk) $100 ($15)
with fig line editor.
** tFORTH + — more! (3 5" or 2 8" disks) $250 ($25)
adds screen editor, assembler, extended data types, utilities, games, and debugging aids.
** TRS-80 COLORFORTH — available from The Micro Works
** firm FORTH — 6809 only. $350 ($10)
For target compilations to rommable code.
Automatically deletes unused code. Includes HOST system source and target nucleus so rce. No royalty on targets. Requires but does not include tFORTH +.
** FORTH PROGRAMMING AIDS — elaborate decompiler $150

** tFORTH for HX-20, in 16K roms for expansion unit or replace BASIC $170

** tFORTH/68K for CP/M-68K 8" disk system $290
Makes Model 16 a super software development system.

** Na til s Systems Cross Compiler
— Requires: tFORTH + HOST + at least one TARGET:
— HOST system code (6809 or 68000) $200
— TARGET source code: 6800-$200, 6301/6801—$200
same plus HX-20 extensions— $300
6809—$300, 8080/Z80—$200, 68000— $350

Manuals available separately — price in ( ).
Add $6 system for shipping, $15 for foreign air.

TALBOT MICROSYSTEMS 1927 Curtis Ave., Redondo Beach, CA 90278 (213) 376 9941

!!! FREE !!!

Published Monthly by Computer Publishing Inc., Hixson, TN. $1.95

Bulk Rate
U.S. Postage
PAID
Chattanooga, TN
Permit No. 357

Color Micro Journal

The Color Computer Monthly Magazine $1.95 per issue Vol. 1, Issue 2 October, 1983

THIS 'N THAT

The BIG NEWS this month is that OS-9 has finally arrived for the Color Computer. The ASTOUNDING part of the Radio Shack OS-9 Package, besides the price, is the DOCUMENTATION. You 'Old Time Radio Shack Followers' will not believe what you see. Jon Shirley has been telling us that the main reason for the "lack" of documentation with a lot of their products was the restrictions placed on releasing that information by Microsoft; I

OS-9 on the COLOR COMPUTER

One of the "Operating Systems of the Future" is now available for the "little old Color Computer"; OS-9. Freely translated, OS-9 means "Operating System for the 6809" (OS-9 is now being written for the 68000, also). Since it is fairly obvious that UNIX and "UNIX-Type" Operating Systems will be running on just about every computer to come out in the next few years, a whole new language is beginning to appear on the horizon.

Color Computer OS-9; the Package

We had been running a preliminary release of OS-9 on the Color Computer for a few weeks, and received the "Official Radio Shack" version for Review a couple of days ago. To put it mildly, this package is IMPRESSIVE! For $69.95 (Radio Shack Catalog Number 26-3038), you receive a 9 1/2" x 7 5/8" x 2" package containing 4

FREE SAMPLE ISSUE

1-800-338 6800

MON.-FRI. 9-5 E.S.T.

USA-$12.50 per year. Canada& Mexico-$19.50 per year

Surface Foreign-$24.50 per year. Airmail Foreign-$48.50 per year

Color Micro Journal™

TM Color Micro Journal is a trademark of Computer Publishing Inc.

5900 Cassandra Smith Rd. Hixson,TN. 37343

6809 Word Processing System

stylograph™

SPELLING CHECKER
Checks all words against an internal user-expandable dictionary of over 42,000 words.
OS9, FLEX $145 UniFLEX $195

STYLOGRAPH 2.0
The "User Friendly" word processing system. Fewer key strokes by the operator make it easier to learn.
OS9, FLEX $295 UniFLEX $395
COLOR COMPUTER FLEX $195

MAIL MERGE
Inserts names and addresses into form letters and mailing lists. Appends files at print out time. Handles files longer than memory.
OS9, FLEX $125 UniFLEX $175

Inquire about our other software
- Business Programs - G/L, A/R, A/P
- Data Base Management System
- Assemblers

Also, Daisy Wheel Printers $599.

Great Plains Computer Company Inc.
P.O. Box 916
Idaho Falls, Idaho 83401

(208) 529-3210

Flex and Uniflex are trademarks of Technical Systems Consultants, Inc.
OS9 is a trademark of Microware.

MACROPLEX Software

175 Fifth Avenue, Suite 3011, New York, NY 10010

Presenting... **THE CONDUIT**
A new File Type for OS9

THE CONDUIT lets programs exchange data without using external files. Direct program-to-program data transfers on any path use OS9 I/O calls. No need to change device-independent programs. Two-way transfers are permitted and device-dependent code can be accommodated. Programs that formerly ran sequentially can now run concurrently to the limit of memory.

Utilities supplied with **THE CONDUIT** give immediate results when linked to your current programs via **CONDUIT** paths:
- Fast disk slowdown speeds up assemblies and other I/O bound procedures.
- "In-flight" terminal I/O redirection (e.g., set traps in DEBUG from a disk file).
- Basic09 source inclusions, enhanced listing.
- "Null files" drain output, signal EOF on input.

Free brochure available. Write, or call (212) 686-3036.

Manual alone (50 pages)	$ 8
CONDUIT file system + utilities	$70
Both together	$75
Postage outside North America, add	$ 5

Specify 5" or 8" disk

Version for **OS9 Level Two** available soon

OS9,™ Basic 09,™ Microware Systems Corp. & Motorola, Inc.

DATA ACQUISITION CP/M 2.2

NOW THERE IS AVAILABLE ON THE SS50 BUS INDUSTRIAL QUALITY BOARDS FOR YOUR DEMANDING DATA ACQUISITION NEEDS

ADC1200
- *HIGH SPEED 12 BIT A/D BOARD*
- *16 CHANNELS Single-ended or Eight Differential*
- *25 uSEC CONVERSION TIME*
- *80k SAMPLES PER SECOND in single Channel Burst Mode*
- *INSTRUMENTA ION AMPLIFIER / Resistor Selectable Gain*
- *Contained on Single 30 Pin Board*
- *CONFIGUREABLE in a Variety of Computer Controlled Modes*
- *SOFTWARE EXAMPLES*
- *$795 Each $676.2 to 4*

DAC 1220
- *HIGH SPEED 12 BIT D/A BOARD*
- *TWO INDEPENDENT DIGITAL TO ANALOG CONVERTERS*
- *10 uSEC SETTLING TIME*
- *DOUBLE BUFFERED*
- *BLANKING OUTPUT PULSE*
- *FOUR QUANDRANT MULTIPLY using EXTERNAL REFERENCE Input*
- *Contained on Single 30 Pin Board*
- *$395 Each $336.2 to 4*

GPIB4800
- *IEEE 488 CONTROLLER BOARD*
- *Talker, Listener, Controller, Master*
- *Uses the TI 9914 Controller Chip*
- *IEEE 488 Panel Mount Connector*
- *Contained on Single 30 pin Board*
- *$295 Each 1 to 4*

Z809
- *CP/M 2.2 OPERATING SYSTEM*
- *Z80 CO-PROCESSOR*
- *ASSEMBLER DEBUGGER UTILITIES*
- *PUBLIC DOMAIN SOFTWARE*
- *$595 Each $476.2 to 4*

WRITE OR CALL TODAY FOR COMPLETE DATA

FOREIGN DEALERS
Kline Computers • D-5521-Irrel, West Germany
Tel: 652-5299
Digicomp AG • Zurich, Switzerland
Tel: 1-461-12-13
Bernstein Computer Consultants • Cape Town, So. th Africa • Tel: 21-8394

META LAB

(303) 449-1711
6825 COUNTY LINE ROAD 1
LONGMONT, CO 80501

OS-9®
SOFTWARE TOOLS!

- **ADLIB 50.00**
  Allows shared source code
- **ISAM 350.00**
  Indexed Sequential File Access Method
- **SMATH 95.00**
  String Arithmetic
- **SORTC 150.00**
  Full-record disk sort
- **EXAMOD/CHGREV 50.00**
  Determines contents of a module
- **IN-DATA/REPORT-GEN/MENU 75.00**
  Data entry/Menu & Report generator
- **LOOKUP/SLOOKUP 75.00**
  "Wild Card" directory searcher
- **VID 75.00**
  Display/input data edit package
- **XRF 200.00**
  Low-overhead database emulator
- **TERMINAL 95.00**
  Communicates with other machines
- **LDMAC 75.00**
  Assembly code routines "tool box"

For more information, or to place an order, contact:
Dept. 68 8
The JBM Group, Inc.
Continental Business Center
Front & Ford Streets
Bridgeport, PA 19405
Tel: 215-337-3138/TWX: 510-660-3999

the JBM group

VISA/MC accepted: PA residents please add 6% sales tax
U.S. orders please add 5.00 postage/handling

* Registered trademark of Microware

STAR-DOS LEVEL I

Whenever a new DOS is introduced, there's always the problem of developing software to work with it. So we did it the opposite way — we analyzed the requirements of software that already exists and developed a DOS that met them... and exceeded them! The result is STAR-DOS Level I, a new DOS for 6809 systems, ideal for single-user industrial, control, and advanced hobbyist applications. This includes SS-50 systems and single-board computers from a variety of vendors.

Level I is compatible with most current 6809 hardware and software. On the hardware side, it allows up to ten floppy or Winchester drives with appropriate controllers. On the software side, it runs existing 6809 software from all the major 6809 software suppliers, including TSC, Star-Kits, Introl, and others.

Write or call for more information. STAR-KITS Software Systems Corporation. P.O. Box 209, Mt. Kisco N.Y. 10549 (914) 241-0287.

STAR-KITS

ANDERSON COMPUTER CONSULTANTS
&
Associates

Ron Anderson, respected author and columnist for 68 MICRO JOURNAL announces the **Anderson Computer Consultants & Associates,** a consulting firm dealing primarily in 68XX(X) software design. Our wide experience in designing 6809 based control systems for machine tools is now available on a consultation basis.

Our experience includes programming machine control functions, signal analysis, multi-axis servo control (CNC) and general software design and development. We have extensive experience in instrumentation and analysis of specialized software. We support all popular languages pertaining to the 6809 and other 68XX(X) processors.

If you are a manufacturer of a control or measuring package that you believe could benefit from efficient software, write or call Ron Anderson. The fact that any calculation you can do with pencil and paper, can be done much better with a microcomputer. We will be happy to review your problem and offer a modern, state-of-the-art microcomputer solution. We can do the entire job or work with your software or hardware engineers.

Anderson Computer Consultants & Associates
3540 Sturbridge Court
Ann Arbor, MI 48105

OS-9®
SOLUTIONS for your CoCo!

For more information or to place an order, contact:

Dept. 689

The JBM Group, Inc.
Continental Business Center
Front & Ford Streets
Bridgeport, PA USA 19405
TEL: 215-337-3138
TWX: 510-660-3999

the JBM group

VISA/MASTERCHARGE accepted. PA res. add 6% sales tax.
US orders, add $5.00 postage and handling.

* OS9 is a registered trademark of Microware Corp.

- *powerful problem solvers*
- *mainframe computing for your micro*
- *total systems support*

TIME IS RUNNING OUT !

Don't Be The Last One To Subscribe To Our

FREE

Newsletter/Catalog

"Serious Users Software Catalog"

FRANK HOGG LABORATORY

THE REGENCY TOWER • SUITE 215 • 770 JAMES ST. • SYRACUSE, NY 13203
PHONE (315)474-7856 • TELEX 646740

ARCADE 50

POWERFUL COLOR GRAPHICS
Uses the new TMS9918A Video Display procesor. High resolution 256 x 192 pixel display with 15 colors. 16K Bytes of onboard RAM does not reduce user memory. 32 graphic images can be individually moved with simple X-Y commands for smooth animation. External Video input allows subtitling. NTSC composite video output.

SOUND EFFECTS AND MUSIC
- Three AY3-8910 Programmable Sound General s
- Nine simultaneous voices
- Three independent noise sources
- Onboard stere amplifier drives lw 8 ohm speakers

ADDITIONAL I/O CAPABILITIES
- Eight analog inputs with 8 bit resolution
- Supports four joysticks with pushbutton switches
- Eight bit parallel I/O port
- Entire unit maps into 256 bytes of memor

FBASIC

TERMINUS DESIGN INC. in conjunction with Microware Systems Corporation, is proud to announce FBASIC an enhancement of Microware's 6800/BASIC. Their fast compiled BASIC has been adapted for 6809 users with added video and sound features for ARCADE 50 users. FBASIC is a true compiler that produces optimized machine language modules which are ROMable and require no Run-Time package. FBASIC requires less memory overhead and runs hundreds of times faster than BASIC interpreters. It supports standard BASIC instruction including String functions. Disk I/O and fast integer arithmetic with multiple-precision capability. Graphics verbs and functions fully support the Arcade 50.

ARCADE 50 assembled and tested	$325.00
Video and Audio connector set	15.00
4 Joystick c nnector set	15.00
2 Radio S ack Joysticks	24.00
Gold Molex connectors	12.00
A/BASIC for 6800	110.00
FBASIC for 6809	110.00
FBASIC (with ARC DE 50)	75.00
ARCADE 50 RGB	375.00
LABVIDEO (Motorola ExORbus)	375.00
NEW MV09 6809 Processor Board	225.00
256K Dynamic Memor Board	795.00
256K Dynamic Memory Board (w/64K)	395.00
64K Dynamic Mem y Board	295.00

TERMINUS DESIGN INC
16 SCARBROUGH ROAD
ELLENWOOD, GA 30049
(404) 474-4866

TERMS: CASH, VISA, MC, C.O.D.

Computer Servo Controlled Robot Arm

Call or Write for Free Catalog

Analog Micro Systems

5660 Valmont Road - Boulder, Colorado 80301 - Tel: (303) 444-6809

Robot-1

Keyboard or Joystick Control

Remembers Everything It Did & does it again

Typical System Includes:
- Robot-1 & Cables
- 6 Channel Servo Controller
- Power Supply
- All Software with source code

Modular Robotic Accessories:
- Mobile Cart for Traveling Robot
- Radio Links between all Functions
- Robot-mounted MicronEye
- Ultrasonic Range Finder

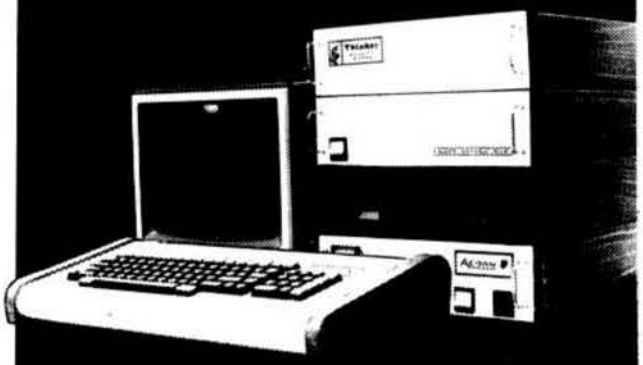
Robot-1 Series

starting at $289.00
for the Color Computer
and 6809 SS50 Computers

Additional Systems Available

Robot-1R for Radio Control Systems

ACORN
COMPUTER SYSTEMS SS-50C



MODULES - BARE CARDS - KITS - ASSEMBLED & TESTED

Stackable Modules		KIT	A&T
20 amp POWER SUPPLY w/fan w/Disk protect relay		350.00	400.00
DISK CABINET w/rege. & cables less DRIVES		200.00	250.00
MOTHER BOARD, 6 SS-50c, 6 SS-30c N I button		225.00	325.00
Item	**Bare**	**KIT**	**A&T**
IT3 - INTERRUPT TI ER 1, 10, 100 per sec.	19.95	29.95	39.95
P84 - INTELLIGENT PORT BUFFER Single board comput.	39.95	114.95	139.95
DPIA - Dual PIA parallel port, 4 buffered I/Os	24.95	69.95	69.95
XADR - Extended Addressing BAUD gen. PIA port	29.95	69.95	69.95
MB8 - MOTHER BOARD SS-50c w/BAUD gen.	64.95	149.95	199.95
P16E - 16EK PROM DISK 21, 2764 EPROMs	39.95	79.95	109.95
FD8E - Firmware development 2, 8K blocks	39.95	84.95	114.95
EXPR - 2764 PROM burner adapt. for 2716 BURNER		19.95	------
CHERRY Keyboard w/Cabinet 96 key capacitive		249.95	------
TAXAN 12", 18 hz MONITOR GREEN		------	149.95
AMBER		------	159.95
4 MODULE CABINET - unfinished		150.00	------
POWER SUPPLY w/disk protect		250.00	------

Color Computer

MONOLINK - 20 Mhz Monochrome video driver	15.00	20.00
CC3G PORT BUS e/power supply 5 SS-30, 2 Cart	169.95	199.95
POWER BOX 6 switched outlets transient suppression	29.95	39.95
RS-232 3-switched ports for above	ADD +20.00	+25.00

Write for FREE Catalog
ADD $3.00 S&H PER ORDER
WIS. ADD 5% SALES TAX

11931 W. Bluemound Road
MILWAUKEE, WIS. 53226
(414) 257-0300

68' MICRO JOURNAL ADVERTISERS INDEX

'68' MICRO JOURNAL 49,60
AAA CHICAGO COMPUTER CENTER 36,37
ACORN COMPUTER SYSTEMS 70
ANALOG MICRO SYSTEMS 69
ANDERSON COMPUTER CONSULTANTS 67
COLOR MICRO JOURNAL 65
COMPILER EVALUATION SERVICES 49
COMPUTER PUBLISHING INC... 5
COMPUTER SYSTEMS CENTER 59,60
COMPUTER SYSTEMS CONSULTANTS, INC.....64
DATA-COMP .....IBC
DIGITAL RESEARCH COMPUTERS 62,63
FRANK HOGG LABORATORY, INC... 69
GIMIX, INC... 3,72
GREAT PLAINS COMPUTER CO... 66
HAZELWOOD COMPUTER SYSTEMS 51,OBC
INTROL CORP... 50
JBM 67,68
LSI ENTERPRISES LTD... 58
MACROPLEX SOFTWARE 66
META LAB 66
MICROWARE SYSTEMS CORP... 1,4,13
PERIPHERAL TECHNOLOGY 48
ROBERTSON ELECTRONICS 58
SMOKE SIGNAL BROADCASTING 6,7
SOUTH EAST MEDIA 52,53,54,55,56,57
SOUTHWEST TECHNICAL PRODUCTS INC. ...IFC
STAR-KITS 67
TALBOT MICROSYSTEMS 64
TERMINUS DESIGN, INC... 69
THOMAS INSTRUMENTATION 71
UNITEK 58
WESTCHESTER APPLIED BUSINESS SYSTEMS .71
WINDRUSH MICRO SYSTEMS LIMITED 61

This index is provided as a reader service. The publisher does not assume any liability for omissions or errors.

THOMAS INSTRUMENTATION

HARDWARE	ASSEM & TEST	BARE CARD
CPU-5	$299.00	$49.00
S-R/R w/o memory chips	$120.00	$49.00
S-R/R with 48K	$399.00	$49.00
SP-1	$195.00	$49.00
6802 SUPER CPU	$235.00	$59.00
VIDEO RAM	$195.00	$49.00
PARALLEL I/O	$139.00	$49.00
SS-50/50C EXTENDER	$ 35.00	NA
SS-30 EXTENDER	$ 25.00	NA
SS-50 WIRE WRAP	NA	$39.00
SS-30 WIRE WRAP	NA	$20.00
SS-30 BACKPLANE 8POS.	NA	$39.00
SS-50 BACKPLANE 4,8,12, & 16 @ $5.00/SLOT	NA	NA

COMPONENTS

GOLD MALE MOLEX $1.60 GOLD FEMALE MOLEX $1.60
TIN MALE MOLEX $ .40 TIN FEMALE MOLEX $ .50
146805E2P $20.00

SOFTWARE

VDISK
Use extended memory as a fast disk drive
6809 source & obj $149.00
6809 object $ 99.00

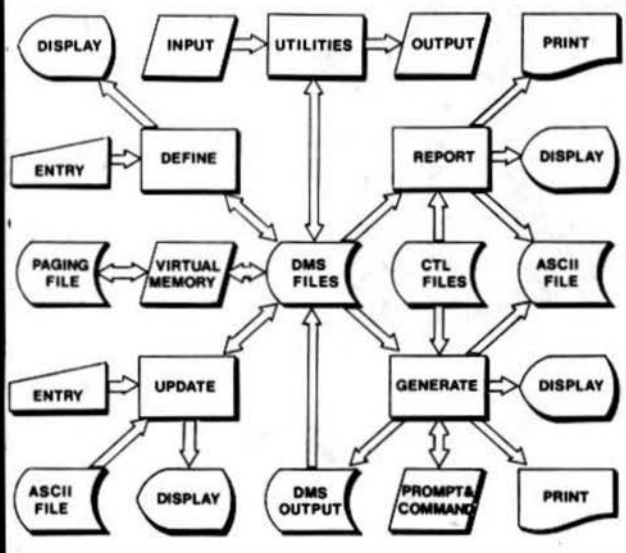
OUTSIDE MODEM PROGRAM
includes source
UniFLEX™ $100.00
FLEX™ 6800 or 6809 $ 50.00

CROSS ASSEMBLER
For 6800, 6801, 6805
Runs on 6809 FLEX™
$150.00

THOMAS INSTRUMENTATION
168 EIGHTH STREET
AVALON, N.J. 08202
(609) 967-4280

MASTERCARD, VISA, AND C.O.D. ACCEPTED. PLEASE ALLOW 3 WEEKS FOR CHECKS TO CLEAR. CONT. USA ADD $3.00 SHIPPING, CANADA $6.00, FOREIGN $12.00. NJ RESIDENTS ADD 6% SALES TAX
FLEX and UniFLEX ARE TRADEMARKS OF TECHNICAL SYSTEMS CONSULTANTS

XDMS
Data Management System



System Architecture

WESTCHESTER Applied Business Systems
Post Office Box 187
Briarcliff Manor N.Y. 10510

XDMS Data Management System
The XDMS Data Management System is available in three levels. Each level includes the XDMS nucleus, VMGEN utility and System Documentation for level III. XDMS is one of the most powerful systems available for 6809 computers and may be used for a wide variety of applications. XDMS users are registered in our database to permit distribution of product announcements and validation of user upgrades and maintenance requests.

XDMS Level I
XDMS Level I consists of DEFINE, UPDATE and REPORT facilities. This level is intended as an "entry level" system and permits entry and reporting of data on a "tabular" basis. The REPORT facility supports record and field selection, field merger, sorting, line calculations, column totals and report titling. Control is via a English-like language which is upward compatible with level II. XDMS Level I $129.95

XDMS Level II
Level II adds to Level I the powerful GENERATE facility. This facility can be thought of as a general file processor which can produce reports, forms and form letters as well as file output which may be re-input to the facility. GENERATE may be used in complex processing applications and is controlled by a English-like command language which encompasses that used by Level I. XDMS Level II $199.95

XDMS Level III
Level III includes all of level II plus a set of useful DMS Utilities. These utilities are designed to aid in the development and maintenance of user applications and permit modification of XDMS system parameters, input and output of XDMS files, display and modification of file format, graphic display of numerical data and other functions. Level III is intended for advanced XDMS users. XDMS Level III $299.95
XDMS System Documentation only ($10. credit toward purchase). . . $ 24.95

XACC Accounting System
The XACC General Accounting System is designed for small business environments of up to 10,000 accounts and inventory items. The system integrates accounting functions and inventory plus the general ledger, accounts receivable and payable functions normally sold separately in other systems. Features user defined accounts, products (or services), transactions, invoicing, etc. Easily configured to most environments.
XACC General Accounting System (Requires XDMS, pref. Lv. III). . $299.95
XACC System Documentation only ($10. credit toward purchase). . . $ 24.95

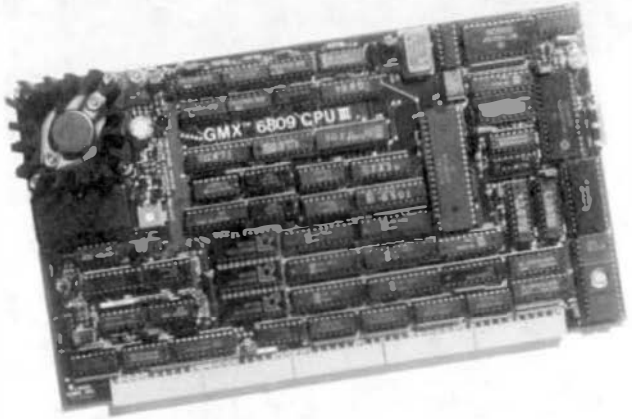
WESTCHESTER Applied Business Systems
Post Office Box 187, Briarcliff Manor, N.Y. 10510

All software is written in micro/assembler and runs under 6809 FLEX O/S. Terms: Check, Money Order, Visa or Mastercharge. Shipment first class. Add P&H $2.50 ($7.50 Foreign). NY Res add sales tax. Specify 5" or 8".

Sales: S. E. MEDIA, 1-800-338-6800, Consultation: 914-941-3552 (evens).

FLEX is a trademark of Technical Systems Consultants, Inc.

GIMIX STATE OF THE ART 6809 SYSTEMS FOR THE SERIOUS USER.



GIMIX has 19MB or high performance 47MB Winchester Drive Systems and/or Floppy Disk Drive Systems.

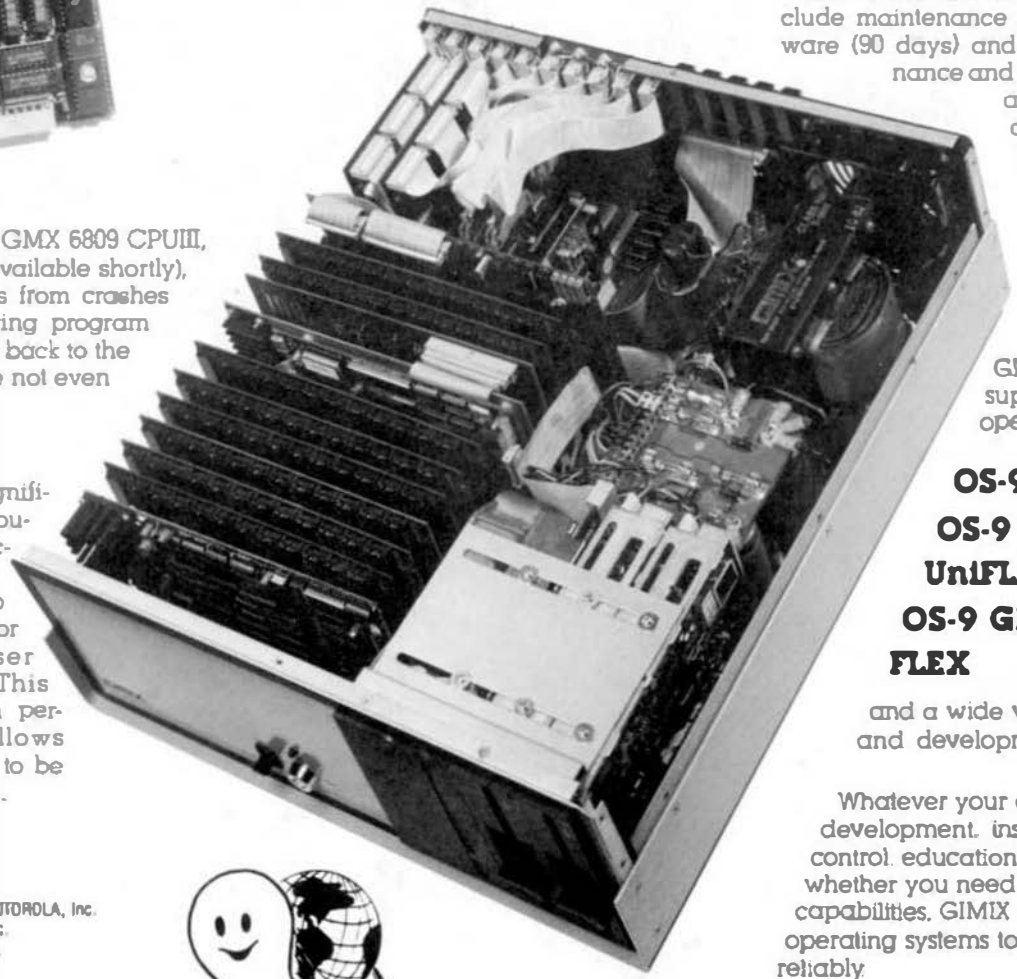
For the user who appreciates the need for a bus structured system using STATIC RAM and powered by a ferro resonant constant voltage transformer.

GIMIX has single user systems that can run both FLEX and OS-9 or Multi user systems for use with UniFLEX or OS-9.

GIMIX versions of OS9 and UniFLEX include maintenance and support by Microware (90 days) and TSC (1 year). Maintenance and support after this period are available at extra cost.

(NOTE: this support and maintenance is only for use with approved GIMIX hardware)

For the ultimate in performance, the Unique GMX 6809 CPUIII, using either OS-9 GMXIII or UniFLEX GMXIII (available shortly), gives protection to the system and other users from crashes caused by defective user programs. e.g. During program development, a programmer who crashes goes back to the shell or the debugger, while the other users are not even aware anything occurred.

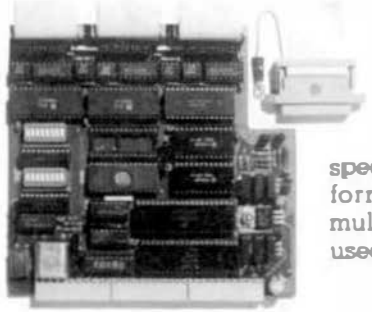


GIMIX 6809 systems support five predominant operating systems

OS-9 GMX III,
OS-9 GMX II,
UniFLEX,
OS-9 GMX I,
FLEX

and a wide variety of languages and development software.

The intelligent serial I/O processor boards significantly reduce system overhead by handling routine I/O functions, thereby freeing up the host CPU for running user programs. This speeds up system performance and allows multiple terminals to be used at 19.2K baud.



Whatever your application: software development, instrumentation, process control, educational, scientific or business, whether you need single or multi-user capabilities, GIMIX has hardware and the operating systems to get the job done reliably.

Please phone or write if you need further information.

BASIC-09 and OS-9 are trademarks of Microware Systems Corp. and MOTOROLA, Inc.
FLEX and UniFLEX are trademarks of Technical Systems Consultants, Inc.
GIMIX, GHOST, GMX, CLASSY CHASSIS, are trademarks of GIMIX, Inc.

GIMIX INC.

1337 WEST 37th PLACE • CHICAGO, ILLINOIS 60609 • (312) 927-5510 • TWX 910-221-4055

©1983 GIMIX Inc.

The Original FLEX™ for Color Computers

- Upgrade to 64K
- RS to FLEX, FLEX to RS file transfer ability
- Create your own character set
- Automatic recognition of single or double density and single or doubled sided
- All features available for either single or multiple drive systems
- Settable Disk Drive Seek Rates
- Faster High Resolution Video Display with 5 different formats
- Save RS Basic from RAM to Disk
- Move RS Basic to RAM
- Load and save function on FLEX disk
- 24 Support Commands 12 with Source Text
- External Terminal Program

DATA-COMP has everything you need to make your TRS-80C Color Computer WORK for YOU! from Parts and Pieces to Full Ready To Use Systems. DATA-COMP designs, sells, services, and SUPPORTS Computer SYSTEMS, not just Software. CALL DATA-COMP TODAY to make your Computer WORK FOR YOU!

System Requirements

FLEX9 Special General Version + Editor & Assembler (which normally sell for $50.00 ea.) $250.00

F-MATE(RS) FLEX9 Conversion Rout. for the RS Disk Controller
when purchased with Special General FLEX9 Sys. $ 49.95
when purchased without the General FLEX9 Sys. $ 59.95

COLOR COMPUTER II WITH 64K AND EXT. BASIC $ 239.95

SPECIAL SYSTEM PACKAGES

64K Radio Shack COLOR COMPUTER, Radio Shack COLOR DISK CONTROLLER, a Disk Drive System, Special General Version of FLEX9, F-MATE(RS), and a Box of 10 Double Density Diskettes, a COMPLETE ready to run SYSTEM on your Color TV Set.

$ 929.85

64K Radio Shack COLOR COMPUTER, Radio Shack COLOR DISK CONTROLLER, a Disk Drive System, and a Box of 10 Double Density Diskettes, a COMPLETE ready to run SYSTEM on your Color TV Set.

$ 629.90

NEW PRICE REDUCTION

DISK SYSTEMS FOR THE COLOR COMPUTER

THESE PACKAGES INCLUDE DRIVE, *CONTROLLER, POWER SUPPLY & CABINET, CABLE, AND MANUAL.

* SPECIFY WHAT CONTROLLER YOU WANT J&M, OR RADIO SHACK.

PAK #1 - 1 SINGLE SIDED, DOUBLE DENSITY SYS.	$389.95
PAK #2 - 2 SINGLE SIDED, DOUBLE DENSITY SYS.	$639.95
PAK #3 - 1 DOUBLE SIDED, DOUBLE DENSITY SYS.	$439.95
PAK #4 - 2 DOUBLE SIDED, DOUBLE DENSITY SYS.	$699.95
PAK #5 - 2 THINLINE DOUBLE SIDED, DOUBLE DENSITY SYS.	$659.95

PRINTERS

EPSON RX-80	$325.00
EPSON RX-80FT	$375.00
EPSON MX-100	$675.00
EPSON FX-100	$749.00
EPSON FX-80	$549.00
EPSON MX-70	$230.00

SERIAL BOARDS

MX-Series	$119.95
RX-FX-Series	$ 99.95

H J L Keyboards $ 74.95

FLOPPY DISKETTES

	MEMOREX	VERBATIM
5" Soft Sector Disks		
Single Sided Single Density	$2.40ea.	$2.75ea
Single Sided Double Density	$2.40ea	$2.75ea
Double Sided Double Density		$4.92ea
Plastic Storage Box		$2.00ea
8" Soft Sector Disks		
Single Sided Single Density		$3.75ea
Single Sided Double Density		$4.10ea
Double Sided Double Density		$4.75ea
Plastic Library Box		$5.00ea

PARTS AND PIECES

Radio Shack Disk Controller	$ 134.95	Single Drive Disk Cable for RS Controller	$ 19.95
1 Single Sided, Double Density Disk Drive Tandon	$229.95	Double Drive Disk Cable for RS Controller	$ 24.95
1 Double Sided, Double Density Disk Drive Qume	$319.95	Micro Tech. Prods. Inc. LOWER CASE ROM Adapter	$ 74.95
1 Qume Thinline Double Sided, Double Density	$249.95	Radio Shack BASIC Version 1.1 ROM or 1.2 ROM	$ 34.95
1 Tandon 48 Track SSDD with Cabinet and Power Supply	$259.95	Radio Shack Extended Basic ROM	$ 39.95
Single Drive Cabinet with Power Supply	$ 79.95	Radio Shack Disk Basic ROM 1.1	$ 29.95
Double Drive Cabinet with Power Supply	$109.95	Screen Clean-Clears Up Video Distortion On Your Color Computer	$ 39.95
J & M Disk Controller	$144.95	Set of Eight 64K RAM Chips w Mod Instructions	$ 49.95

For Ordering Call TOLL FREE 1-800-338-6800

LAST OF PRODUCTION!!!

!! LIMITED QUANTITY !!

SWTPC 8212 & 8212(W)
Intelligent Terminals

New & Demo Models

At Discount Prices

Remaining Supply of
SWTPC 8212 CRT Terminals

QUME QVT-102 TERMINAL

MasterCard VISA

DATA-COMP

P.O. Box 794 HIXSON, TN 37343
1-615-842-4601

*FLEX is a trademark of Technical Systems Consultants
*OS9 is a trademark of Microware

Data-Comp--South East Media & 68 Micro Journal Are Divisions of CPI

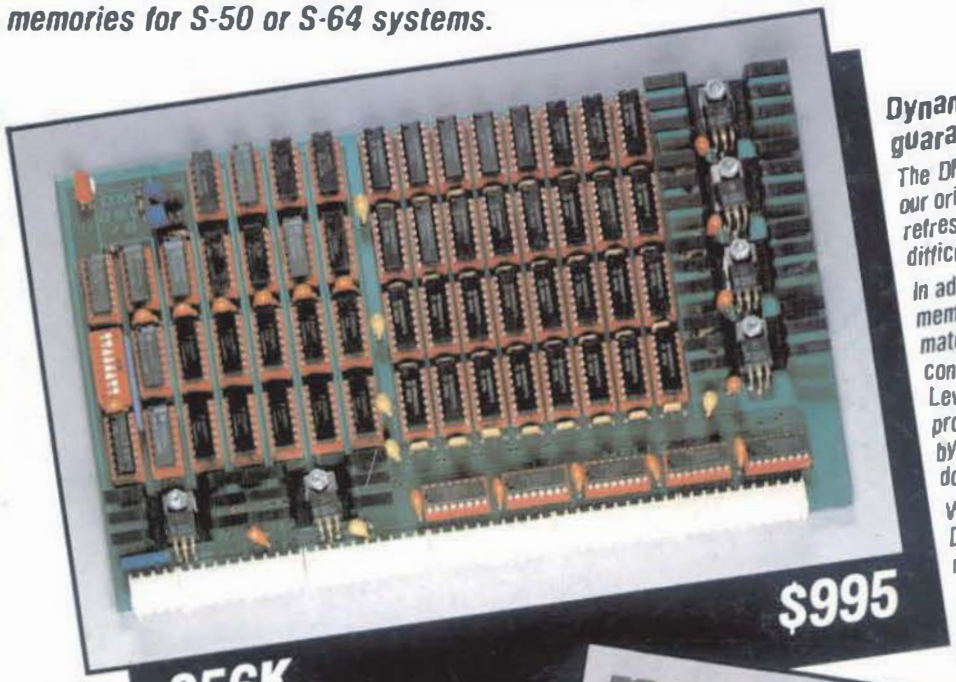
68 Micro Journal
5900 Cassandra Smith Rd.
Hixson, TN 37343

Second Class Postage Paid
Chattanooga, TN
SN 0194-5025

KINGSTON SPRINGS TN 37082
P. O. BOX 87
MR. MICKEY FERGUSON
000422 A/E
MJ

MEMORY • MEMORY • MEMORY • MEM

Hazelwood Computer Systems presents a pair of advanced memories for S-50 or S-64 systems.



256K **$995**

Dynamic memory with guaranteed 2.5 MHz operation

The DM-256 utilizes the same proven refresh method as our original DM-64. This technique allows fully transparent refresh while maintaining full bus speed and eliminating difficult timing problems.

In addition, individual power regulators for each row of memory ICs enhance reliability by providing the ultimate in noise decoupling. Flexible addressing mode controls make the entire 256K available to OS9 Level 2 or UniFLEX. For FLEX, the furnished utility program HYPERDSK,™ formats memory not used by FLEX as a virtual, superfast disk, thus trememdously improving total system performance.

With several hundred already in use, the DM-256 is the best choice for economical, reliable memory.

Assembled, tested, and burned in

Order: DM-256

Static memory with on-board battery backup

Utilizing the latest in low power CMOS static RAMs, the CM-64 has an on-board NiCad battery to maintain memory contents in the absence of bus power. Generous timing margins allow operation at bus speeds in excess of 2 MHz. Zener diode power loss sensing inhibits write operations to prevent data loss during power transitions. Write access can be explicitly controlled by means of an on-board jumper, a remote mounted switch, or an external logic signal.

This allows the memory to appear as ROM to the computer after the contents have been "programmed" using normal computer operations. The economical price makes the advantages of non-volatile memory available to any system.

Assembled, tested, and burned in

Order: CM-64

64K **$395**

HAZELWOOD COMPUTER SYSTEMS

907 East Terra, O'Fallon, MO 63366, 314-281-1055

HELIX™

HYPERDSK is a trademark of Hazelwood Computer Systems • OS9 is a trademark of Microware • FLEX and UniFLEX are trademarks of TSC